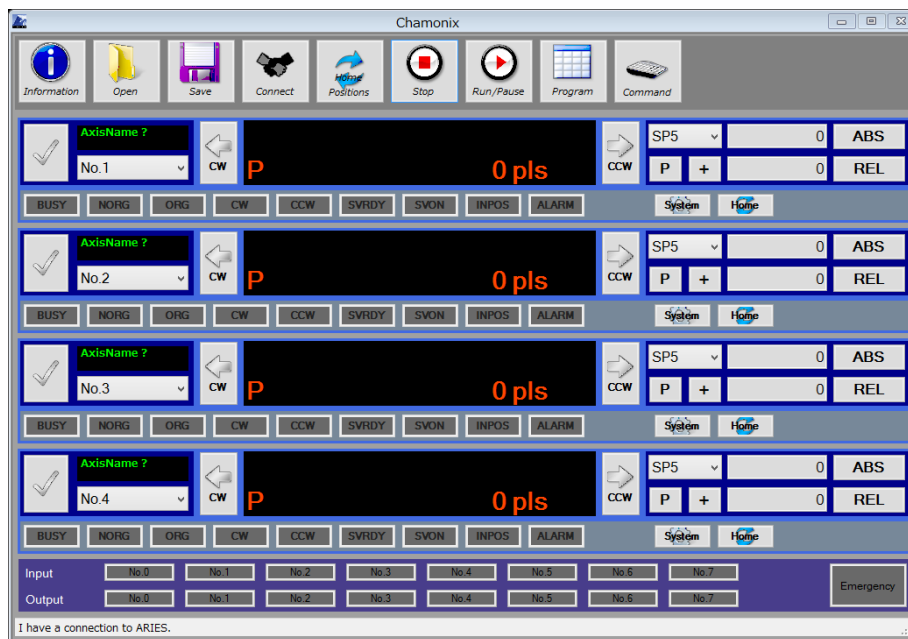




Chamonix

User's Guide

Rev. 1.0.2



Kohzu Precision Co., Ltd.

Table of Contents

Table of Contents.....	3
1. Introduction.....	8
1.1. About Chamonix	8
1.2. About this User's Guide	8
1.3. Precautions for Use	9
1.4. Terms of License Agreement	10
1.5. Supported OS	11
2. Installation	12
2.1. Installation CD.....	12
2.2. Executing Setup.....	13
2.3. Deleting an Old Version	14
2.4. Confirming and Accepting License Agreement.....	15
2.5. Entering User Information.....	16
2.6. Starting Installation	17
2.7. Completing Installation.....	18
2.8. Checking Installation.....	19
3. Uninstallation.....	20
3.1. Control Panel	20
3.2. Uninstalling the Program	21
3.3. Confirmation Dialog	22
4. Wiring the Controller	23
4.1. RS-232C	23

4.2.	Communications Settings of CRUX.....	24
	Rotary switch setting for CRUX communication.....	24
	Positions of rotary switch for CRUX communication setting	25
4.3.	How to Connect Cables on CRUX.....	26
4.4.	Communication Settings of ARIES	27
	Position of Rotary Switch for ARIES Communication setting	27
	Setting of Rotary Switch for ARIES Communication	28
4.5.	How to Connect Cables on ARIES	29
5.	Getting Started!	30
5.1.	Starting from the Icon on the Desktop	30
5.2.	Starting from the Modern UI	31
5.3.	Starting from the Stored File.....	32
5.4.	Start-up Window	33
6.	Connecting the Controller Using RS-232C	34
6.1.	Selecting COM Port	34
6.2.	Connection Check.....	35
6.3.	How to Check the COM PORT Number	36
7.	Checking Version Information	37
7.1.	Checking Version Information about this Application	37
8.	Returning to Origin	39
8.1.	Returning to Origin for All Axes.....	39
8.2.	Status in Returning to Origin.....	40
8.3.	Returning to Origin for Single Axis.....	41

9.	Moving Stages.....	42
9.1.	Selecting a Axis to Control.....	42
9.2.	Entering Axis Names.....	43
9.3.	Switching the Current Position Display.....	44
9.4.	Selecting Speed Table	45
9.5.	Manual Movement	46
10.	Deciding Position	47
10.1.	Setting of Deciding Position with Absolute Position Control	47
10.2.	Executing of Deciding Position with Absolute Position Control	48
10.3.	Setting of Deciding Position with Relative Position Control.....	49
10.4.	Executing of Deciding Position with Relative Position Control	50
10.5.	Executing Multi-Axis Drive	51
11.	Receiving Emergency Stop	52
11.1.	Receiving an Emergency Stop Signal (ARIES)	52
11.2.	Recovering from Emergency Stop (ARIES).....	53
11.3.	For CRUX.....	54
12.	Changing System Parameters	55
12.1.	Starting Parameter Setting.....	55
12.2.	Setting Parameters.....	56
12.3.	Saving System Parameters to Files.....	58
12.4.	Printing Parameters.....	59
13.	Programmable Grid	60
13.1.	Starting Programmable Grid	60
13.2.	Basic Features of Programmable Grid.....	61

13.3.	Running Programs	62
13.4.	Stopping Programs.....	63
13.5.	Inputting and Outputting Program Data	64
13.6.	Commands of Programmable Grid	65
13.6.1.	ORG: Returning to Origin	65
13.6.2.	ABS: Deciding Position with Absolute Position Control	66
13.6.3.	REL: Deciding Position with Relative Position Control	67
13.6.4.	WOT: Controlling General-Purpose Output.....	68
13.6.5.	FOR: Repeat Loop	69
13.6.6.	IF: Running with conditions	70
13.6.7.	GOTO: Jumping to a line.....	70
13.6.8.	SBY: Standby	71
13.6.9.	Command List of Programmable Grid.....	72
14.	Entering Commands Directly.....	73
14.1.	Starting Command Control.....	73
14.2.	Sending Commands from Command Control.....	74
15.	Control from outside using API.....	75
15.1.	Control from outside with Windows API	75
15.2.	Control from External Applications.....	76
15.3.	Sending the APS Command from External Applications	77
15.4.	Sending the RPS Command from External Applications.....	79
15.5.	Sending the STP Command from External Applications	81
15.6.	Sending the ORG Command from External Applications.....	83
15.7.	Canceling Standby of Programmable Grid from External Applications	85

15.8.	Sample Program for External Control	87
16.	Exiting from Chamonix	88
16.1.	Exiting from Chamonix	88
17.	Others	89
17.1.	Folder to save Communication Logs	89
17.2.	Checking Communication Logs	90
17.3.	Detailed Help Files	91
	Index	92
	Revision History	95

1. Introduction

1.1. About Chamonix

Chamonix is an application to control motor controllers from KOHZU Precision. This application can control motor controllers from KOHZU Precision, KOSMOS series (ARIES, LYNX), and old SC series¹ (SC-20, SC-21, SC-200, SC-400, SC-210, SC-410).

Chamonix has GUI to control up to four axes at a time and a programming grid like simplified macro. It offers work with external application through Windows API.

RS-232C serial communication is used to communicate with the controller.

This application may be subject to modify without prior notice.
Read through the terms of the License Agreements carefully for use.

1.2. About this User's Guide

This guide describes Chamonix, an application to control motor controller.

Please read and understand this document thoroughly to utilize the performance and functions of Chamonix in the best condition.

In addition, keep this document in a convenience place for future reference.

Please forward this user's guide to the end user.

¹ Use of SC series may require upgrading of firmware.

1.3. Precautions for Use



Make sure that this application properly works in start of work and operation.



Provide sufficient safety measures to prevent various damage that may arise from accidental failure of this application.



If this product is used in improper conditions other than the terms provided in operation manuals and specifications decided separately, its features and performance cannot be guaranteed.



When this products is used with other devices, use the product after due consideration because the features and performance may not be satisfied depending on conditions of use and environment.



Please do not use to protect a human body.



Avoid the use with other application. Otherwise, features and performance may not be satisfied.

1.4. Terms of License Agreement

Please read this License Agreement carefully before using the Application.

Scope

- This License Agreement applies to Chamonix, a control application supplied as a package; various associated libraries; and all attached documents like manuals describing how to use them. Chamonix, associated libraries, and all attached documents are collectively referred to as the "Application". These documents include all media such as printed materials and electronic files.

Use Rights

- Kohzu Precision Co., Ltd. grants to USER only non-exclusive right to use the Application if you agree to all the terms of this License Agreement. This non-exclusive right allows USER granted the license from Kohzu Precision Co., Ltd. to install and use the Application on a single computer. USER shall not have any other rights except the non-exclusive right. This right does not include charge-free support for users.

Copyright

- Copyrights in and to the Application and all documents with the Application and libraries are owned by Kohzu Precision Co., Ltd. or persons or entities having its applicable copyrights, if not specially provided. The Application is protected by copyright laws and international treaty provisions.

Forbidden Clause

- USER may not resell or distribute the Application, the libraries, or the attached documents to third parties by reproducing them. USER shall not reproduce the Application or the libraries unless for backup purpose. USER may not modify, reverse engineer, disassemble, or decompile the Application. These are strictly prohibited.

Limitation of Liability

- In no event shall Kohzu Precision Co., Ltd. be liable for direct, indirect, or consequential damages of USER and third parties whatsoever arising out of use of the Application. Kohzu Precision Co., Ltd. shall not be responsible for any damage of data, programs, or other intangible property, or any loss of business profits, of USER or third parties.

Others

- Kohzu Precision Co., Ltd. may update the Application at any time. Terms to offer the update shall be decided by Kohzu Precision Co., Ltd. If the Application has been updated, Kohzu Precision Co., Ltd. shall be under no obligation to offer updates compatible with the old application. Kohzu Precision Co., Ltd. may change this License Agreement anytime. If the License Agreement is changed, the former License Agreement shall be nullified and the latest License Agreement applied.

Termination of Agreement

- Kohzu Precision Co., Ltd. shall have the right to immediately terminate this License Agreement if you fail to comply with or do not agree to the terms and conditions of this License Agreement. In such event, you shall return the software and all copies of the software to Kohzu Precision Co., Ltd. or destroy them.

1.5. Supported OS

To run Chamonix, Microsoft.NET4.0 is needed. Chamonix officially support Windows7[®] and Windows8[®].²

WindowsXP[®] is not supported. Note that proper operation cannot be guaranteed for installation in WindowsXP[®].

For installation in WindowsXP[®], system must be installed in the C drive. Severe trouble may cause if system exists in another drive.

Supports both x86 and x64. Installation folders are "Program Files" for x86 and "Program Files(x86)" for x64.

Chamonix is developed by using Microsoft .NET Framework. For installation, you need Microsoft .NET Framework Full 4.5 or later.

² Windows[®] and .NET Framework are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

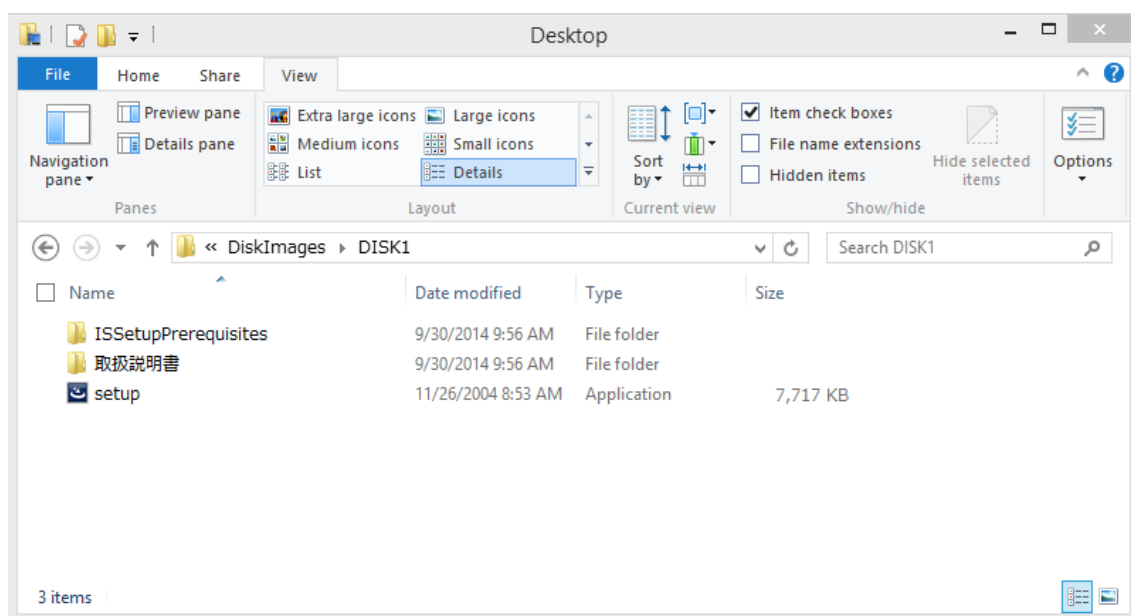
2. Installation

2.1. Installation CD

For installation, you must have administrator privileges. The installation CD consists of files shown below. Click and run Setup to install.

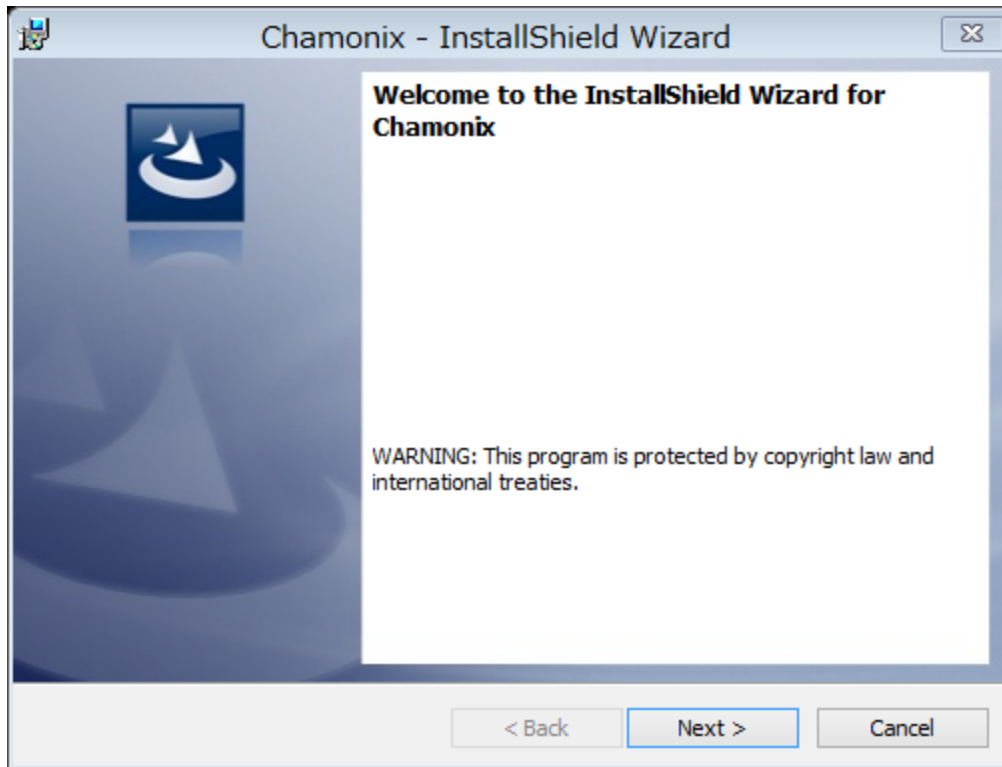
Installation files downloaded from our web site is compressed in a zip file. When you extract the zip file, folders will be created as shown below.

Please do not move and change these folders.



2.2. Executing Setup

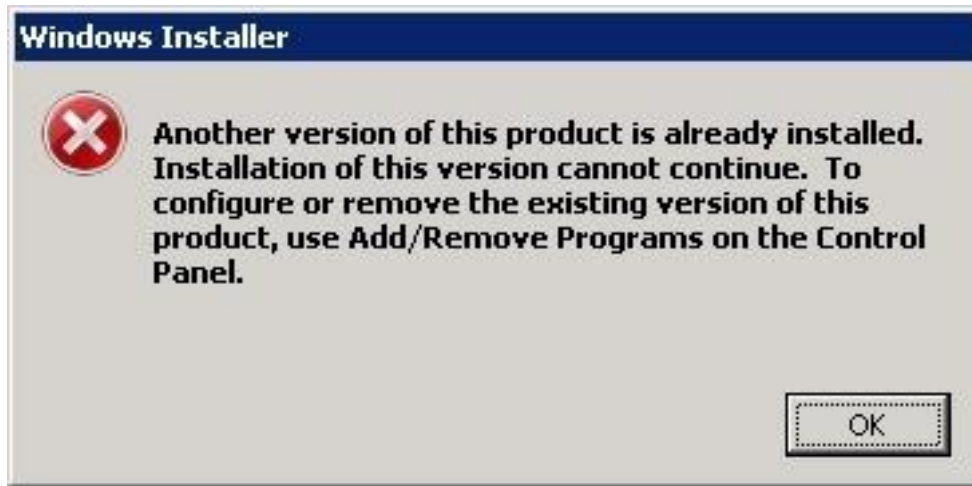
When you execute Setup.exe, the InstallShield^{®3} wizard for Chamonix opens.



Press Next to continue the installation with this wizard. Press Cancel to finish the installation. Then, a window to confirm and accept License Agreement will appear.

³ InstallShield is a registered trademark or trademark of Flexera Software, Inc. and/or InstallShield Co. Inc. in the United States and/or other countries.

2.3. Deleting an Old Version



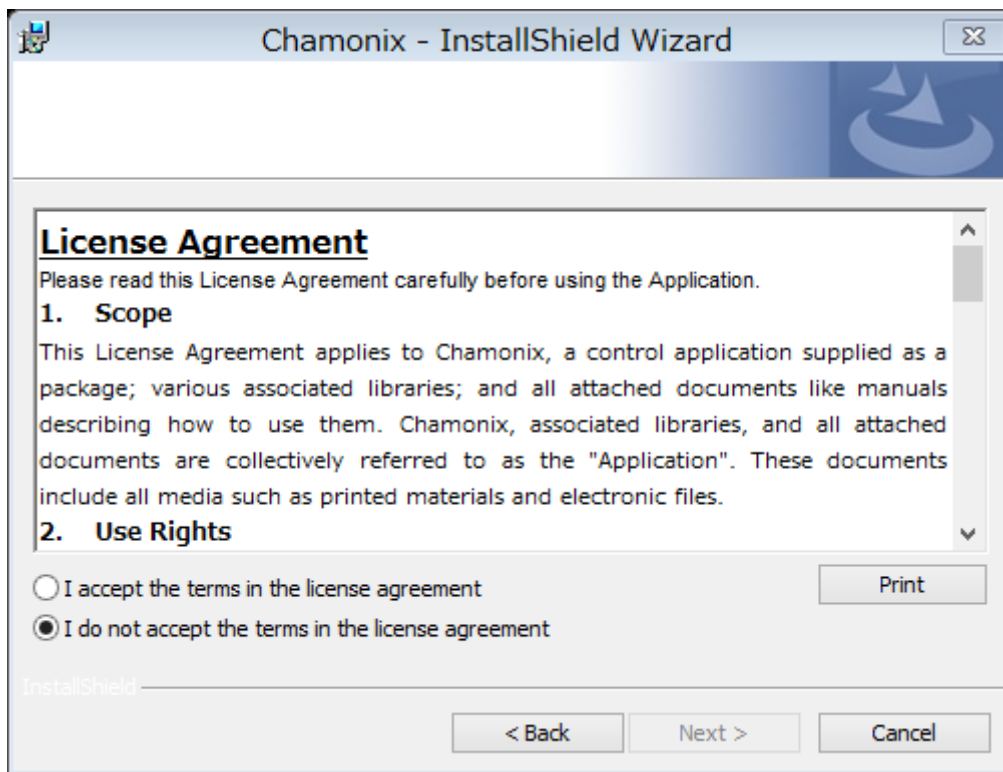
If Chamonix or a part of Dynamic Link Library constituting Chamonix have been already installed, a message⁴ may appear as shown above.

If this message appears, you must remove installed Chamonix using the Add or Remove Programs icon in Control Panel.

Refer to Uninstallation for deletion.

⁴ A window for Microsoft Windows 8.1. A different message may appear depending on OS.

2.4. Confirming and Accepting License Agreement



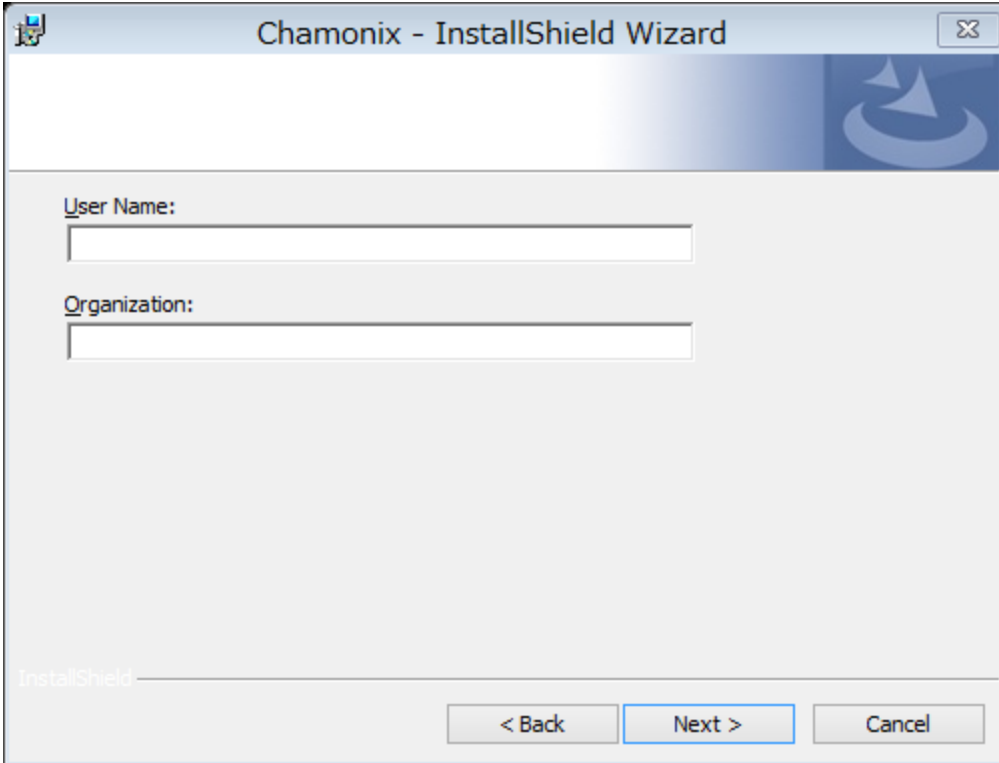
If you agree to the terms of the License Agreement, select "I accept the terms of the License Agreement."

If you do not agree to the terms of the License Agreement, select "I do not accept the terms of the License Agreement." In this case, the Next button is inactive.

Press the Print button to print "The License Agreement".

If you agree to the terms of [the License Agreement](#), press Next to move to Entering User Information.

2.5. Entering User Information



Chamonix - InstallShield Wizard

User Name:

Organization:

InstallShield

< Back Next > Cancel

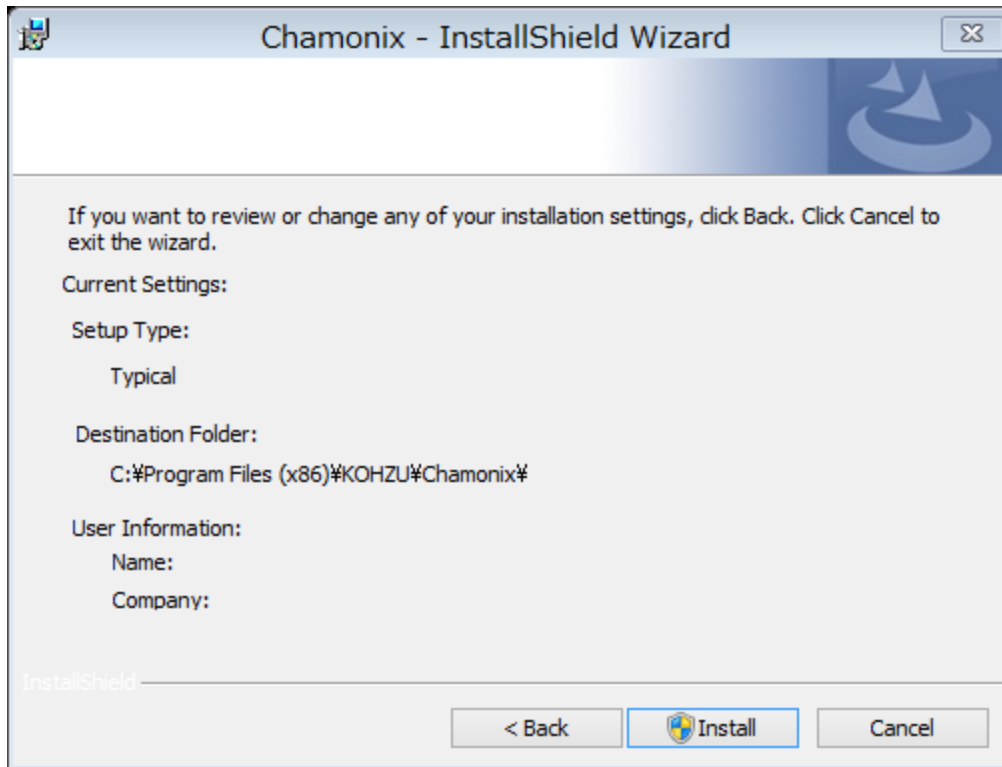
Enter a user name and an organization.

Entering user information is optional.

After entry, click Next. Click Back to check the terms of the License Agreement again.

Click Cancel to cancel the installation.

2.6. Starting Installation



Press Install to start installation.

For installation, you must have administrator privileges.

A user account control dialog may open and ask for starting installation during installation. In this case, check our code signing certificate of the Windows installer and then press Yes.

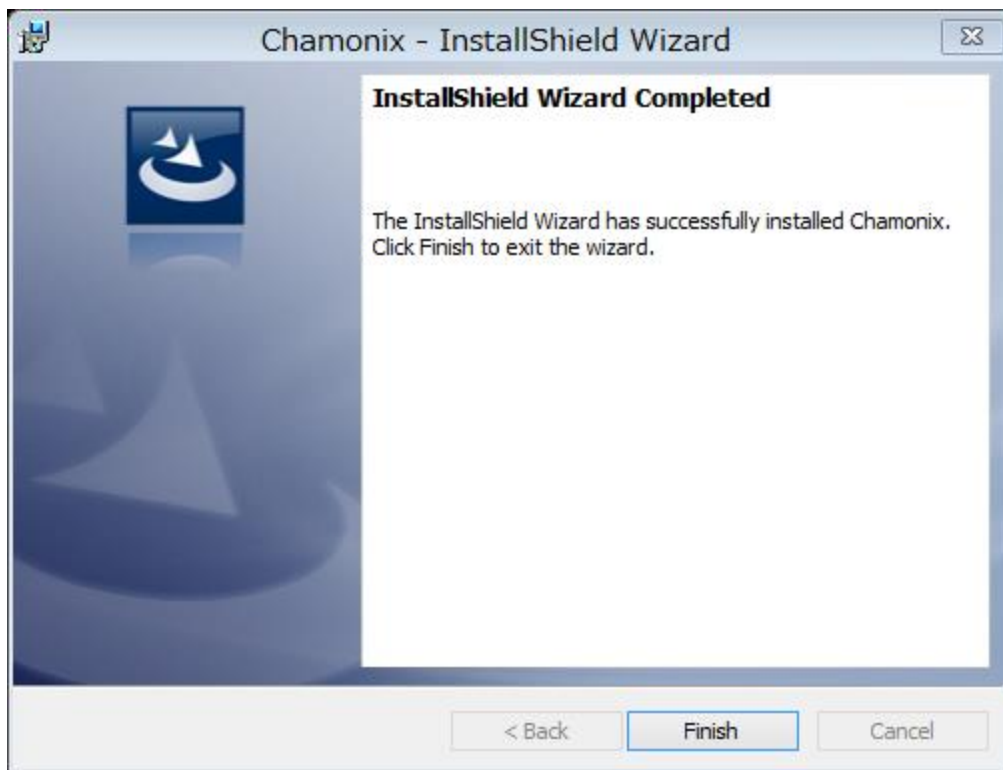
Press No to stop the installation.

The destination folder for installation can not be changed.

Press Back to back to Entering User Information .

Press Cancel to cancel the install wizard.

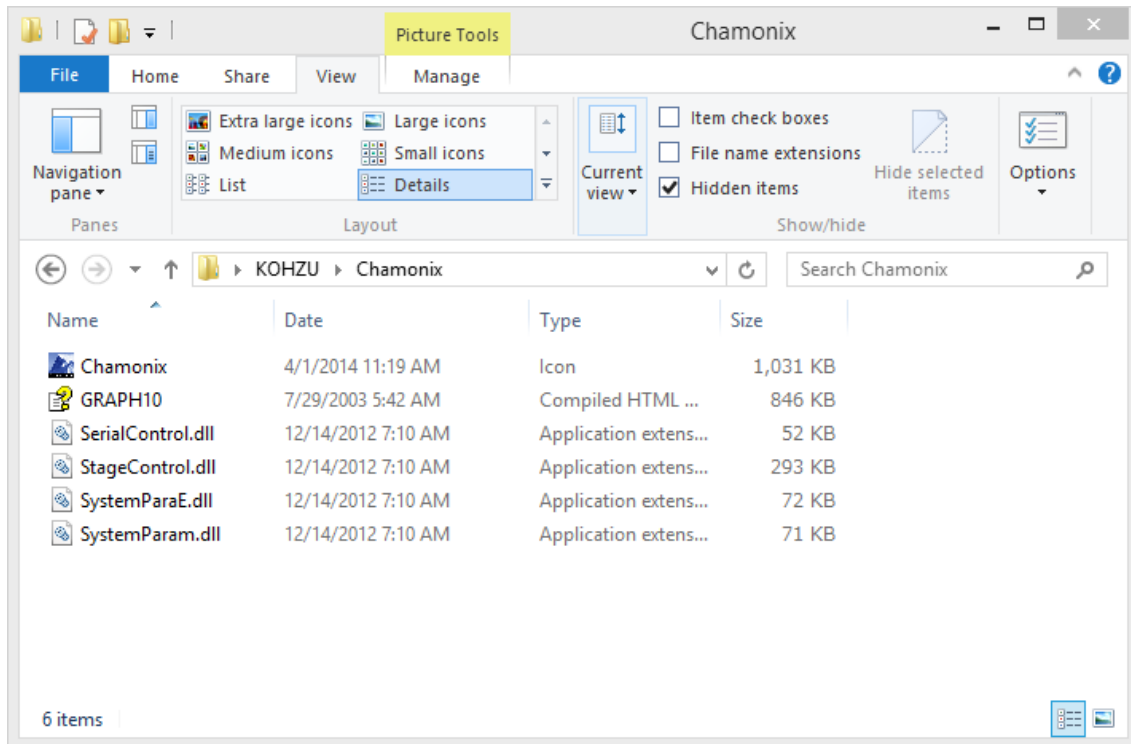
2.7. Completing Installation



Press Install in the previous window to start installing Chamonix. If the above window appears, installation is successfully complete.

Click Finish to finish the wizard.

2.8. Checking Installation

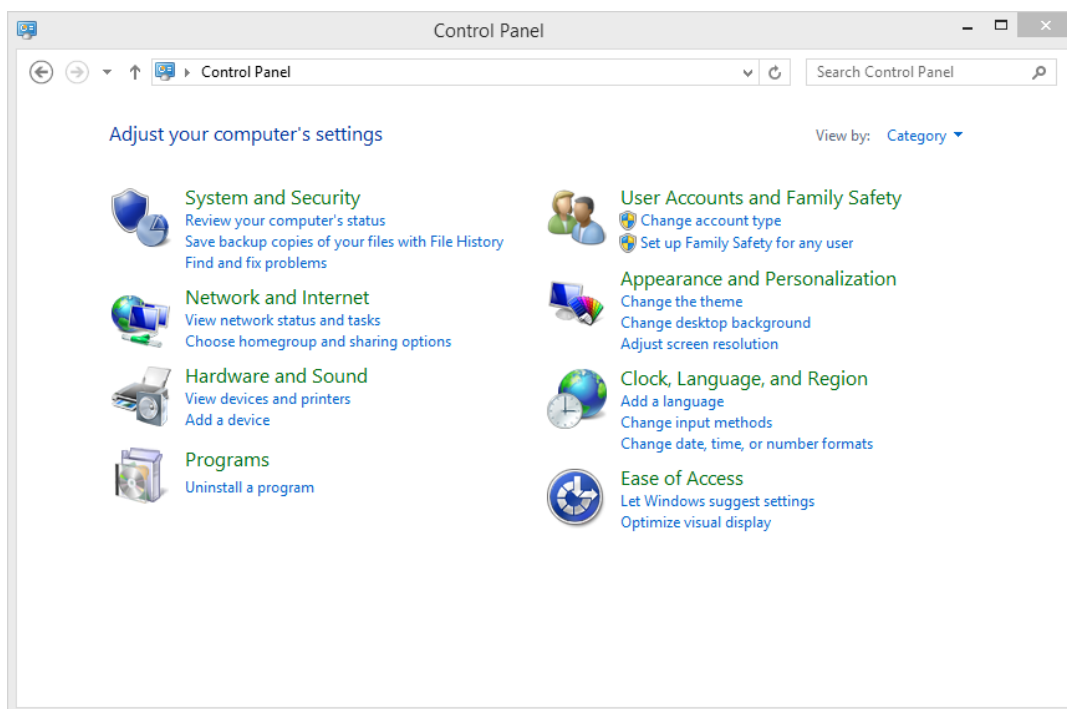


When the installation finished, the six files will be generated in the C:\Program Files(x86)\KOHZU\Chamonix folder as shown above⁵.

⁵ A window for Microsoft Windows 8.1. A different message may appear depending on OS.

3. Uninstallation

3.1. Control Panel



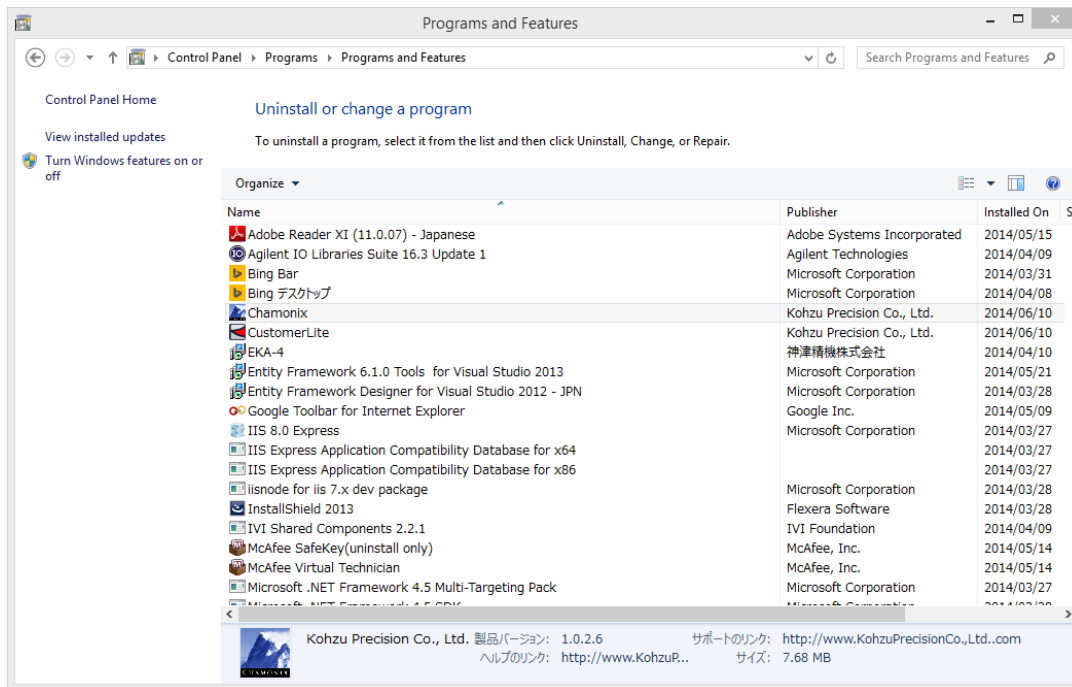
Open Programs in Control Panel⁶. Then click Programs and Features.

Or click Uninstall a program directly in Control Panel.

The Uninstall or Change a program window opens as shown in the next page.

⁶ The above illustration shows an example of Control Panel in Microsoft Windows 8.1.

3.2. Uninstalling the Program



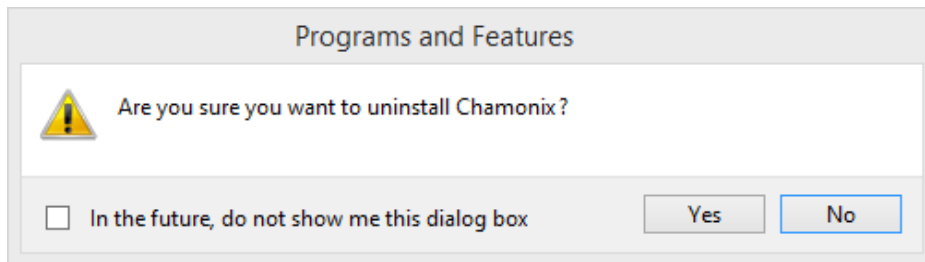
Select Chamonix from the displayed program list.

Click Uninstall to uninstall Chamonix.

A message dialog appears for confirmation.

Click Restore to restore Chamonix.

3.3. Confirmation Dialog



Click Yes in the message dialog to uninstall the program.

Installed Chamonix.exe, StageControl.dll, SerialControl.dll, SystemParam.dll, and Chamonix.chm are deleted in the uninstallation, but SerialContlo_COM*.log and config.bin generated by application after installation are not deleted.

To delete these files, you must delete the files directly with Explorer.

The files are stored in C:\Program Files(x86)⁷\Common Files\KOHZU.

⁷ For a 32-bit OS, the files are in C:\Program Files \Common Files\KOHZU\Chamonix.

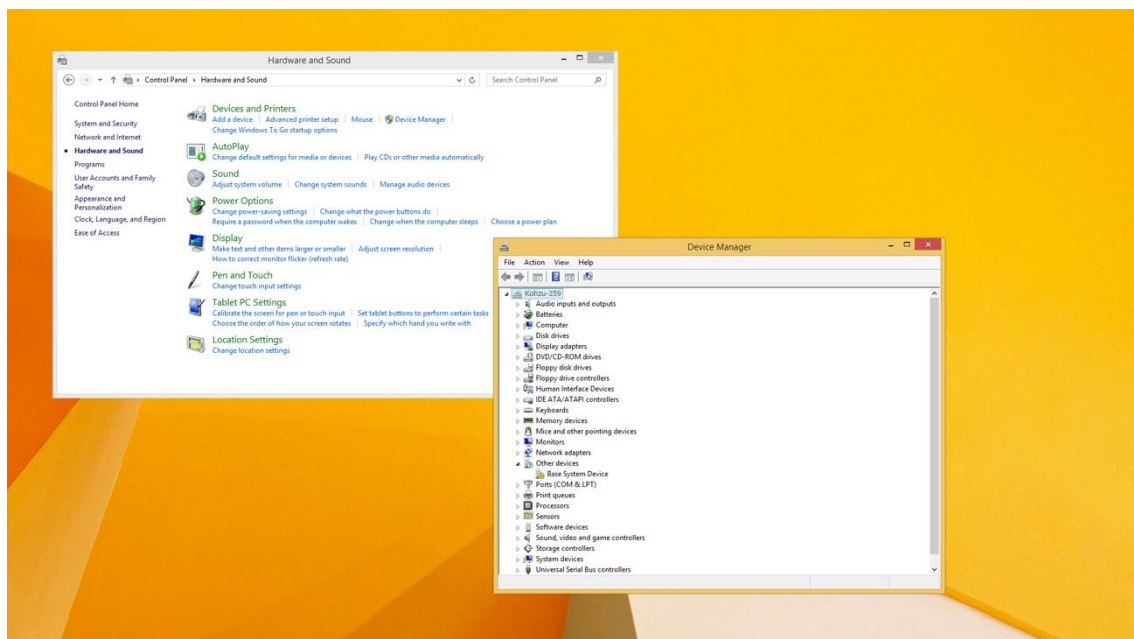
4. Wiring the Controller

4.1. RS-232C

Chamonix supports RS-232C and USB (virtual COM port)⁸.

Check the port connected with Controller in the device manager in advance.

You can find in Control Panel > Hardware and Sound > Device Manager. To start the device manager, you must have administrator privileges.



⁸ The USB virtual COM port is only available for CRUX.

4.2. Communications Settings of CRUX

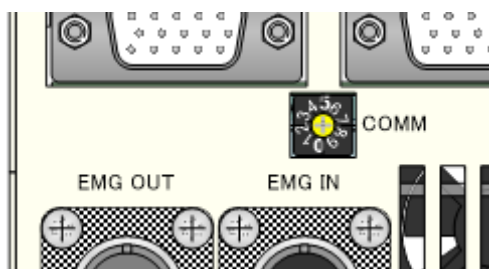
CRUX can set or change communication conditions with the rotary switch (COMM) in the rear panel. By default, the setting is 4 (normal command, USB mode).

See CRUX/CRUX-A User's Manual for more information.

* Settings of RS-232C communication except for speed (baud):

Parity	NON
Word length	8bit
Stop bit	1

Rotary switch setting for CRUX communication



Positions of rotary switch for CRUX communication setting

Communication	Communications settings		
Mode	RS-232C speed	USB	
0	38400	*	General Command
1	57600	*	
2	19200	*	
3	9600	*	
4	*	USB	
5	38400	*	Simple Command
6	57600	*	
7	19200	*	
8	9600	*	
9	*	USB	

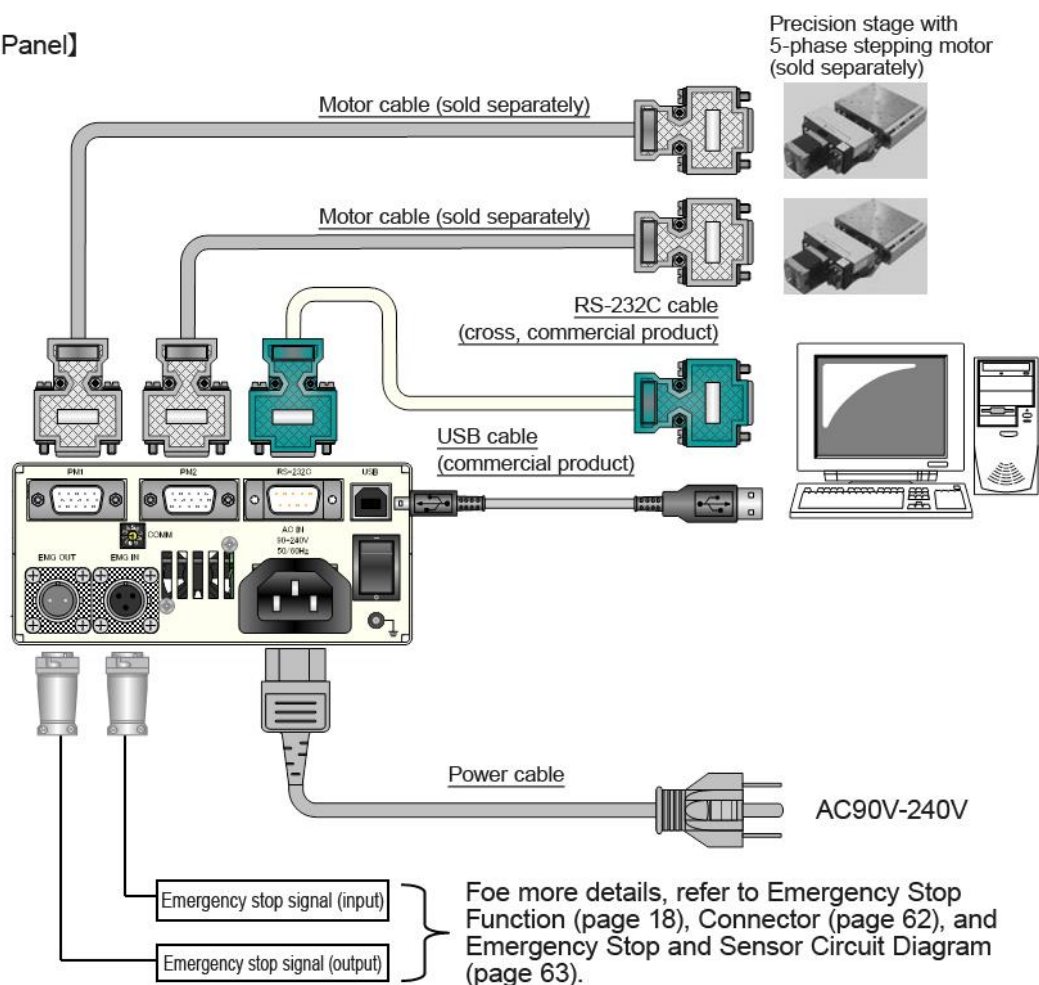
4.3. How to Connect Cables on CRUX



When pulling out or inserting at wire connections, make sure the power of main body is OFF.

Connection/connecting wires between CRUX and external equipment are explained.

【Rear Panel】



See CRUX/CRUX-A User's Manual for more information.

4.4. Communication Settings of ARIES



When configuring communication, make sure the power of main body is OFF.

ARIES can set or change communication conditions with the rotary switch (COMM) in the rear panel. Default setting is Mode 4 (RS-232C 115200baud).

See **CRUX/LYNX User's Manual** for more details.

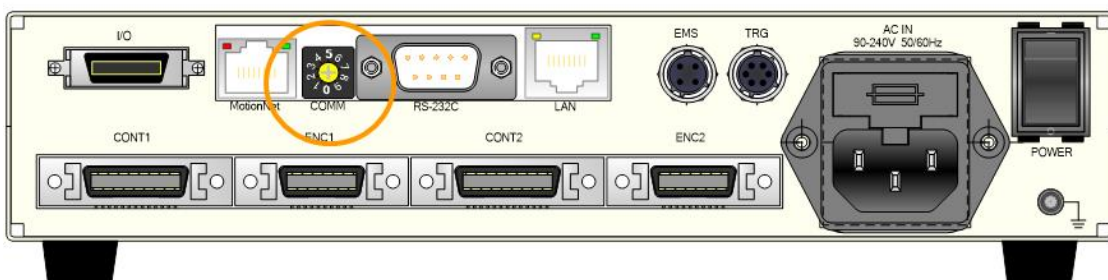
* Settings of RS-232C communication except for speed (baud):

Parity	NON
Word length	8bit
Stop bit	1

Position of Rotary Switch for ARIES Communication setting

The rotary switch for communication setting is located as shown below.

【Rear Panel】



Setting of Rotary Switch for ARIES Communication

Settings are as shown in the table below. (Mode 6 to 9 unusable)

Communication	Communications settings	
Mode	RS-232C speed (baud)	LAN
0	9600	*
1	19200	*
2	38400	*
3	57600	*
4	115200	*
5	*	LAN
6	*	*
7	*	*
8	*	*
9	*	*

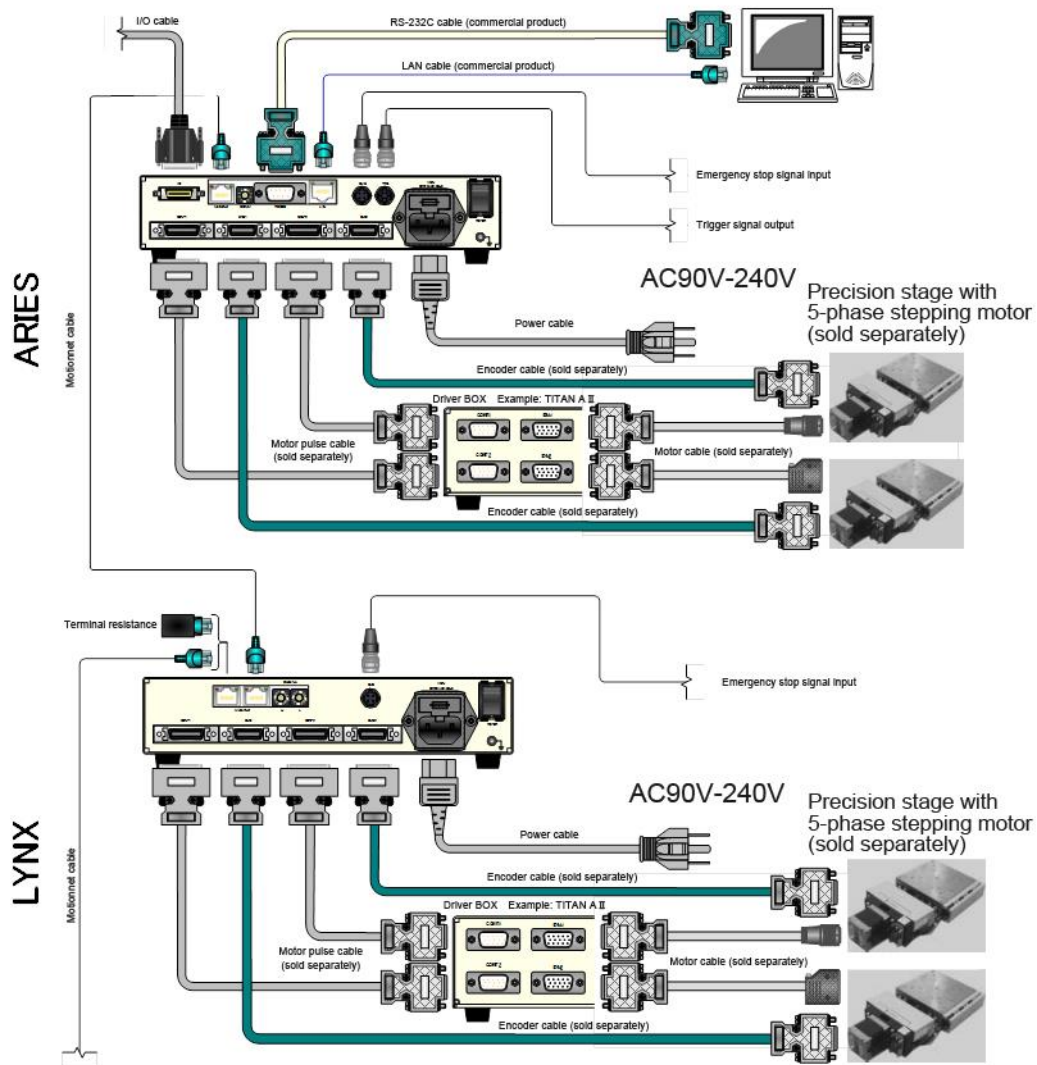
4.5. How to Connect Cables on ARIES



When pulling out or inserting at wire connections, make sure the power of main body is OFF.

Connection/connecting wires between ARIES and external equipment are explained.

【Rear Panel】

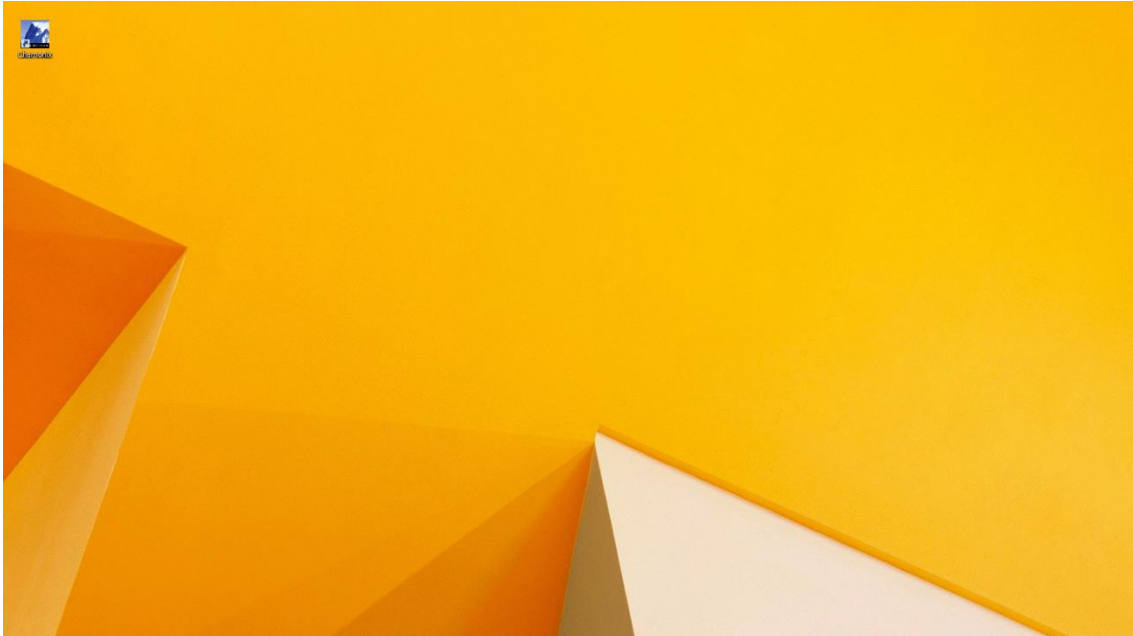


*** Do not use a hub between Motionnet® cable connections.**

See CRUX/LYNX User's Manual for more information.

5. Getting Started!

5.1. Starting from the Icon on the Desktop



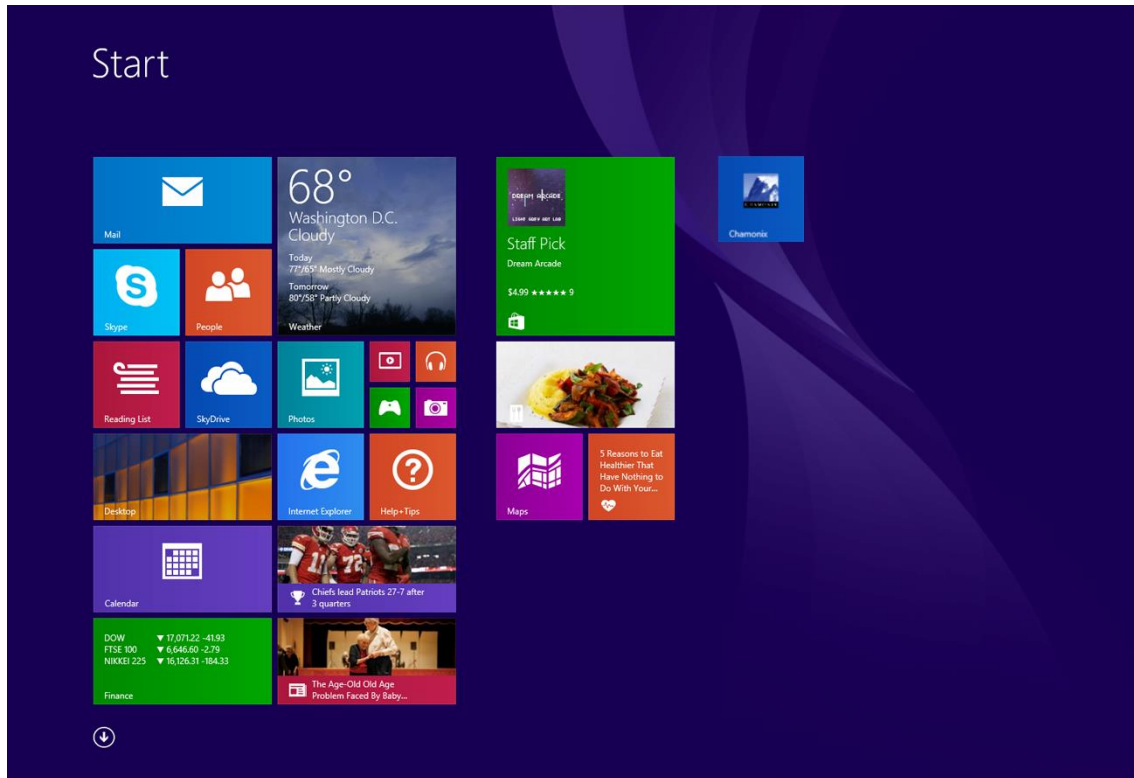
After the installation is successfully complete, the Chamonix icon is generated on the desktop.

Click this icon to start with communication settings that was used last time.

When you start Chamonix for the first time, the default communication settings is used.

By default, the COM port number is COM1 and the baud rate is 115200baud.

5.2. Starting from the Modern UI

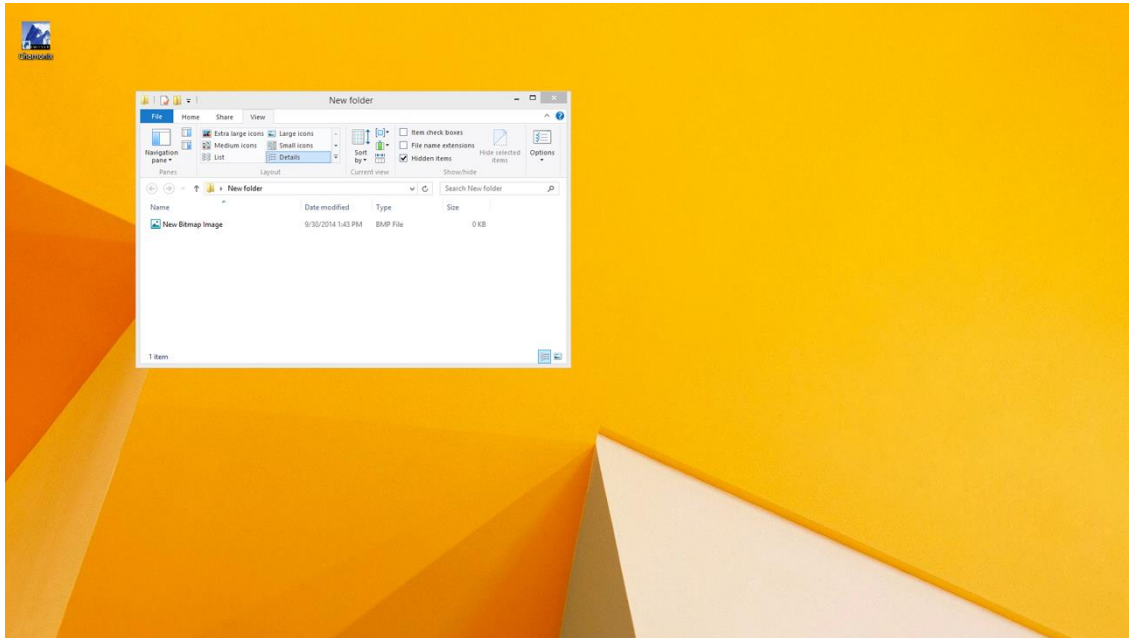


It is also possible to start from the modern UI for Window 8 or later.

It starts with the last parameters like start-up from the desktop. When you start Chamonix for the first time, the default communication settings is used.

By default, the COM port number is COM1 and the baud rate is 115200 baud.

5.3. Starting from the Stored File

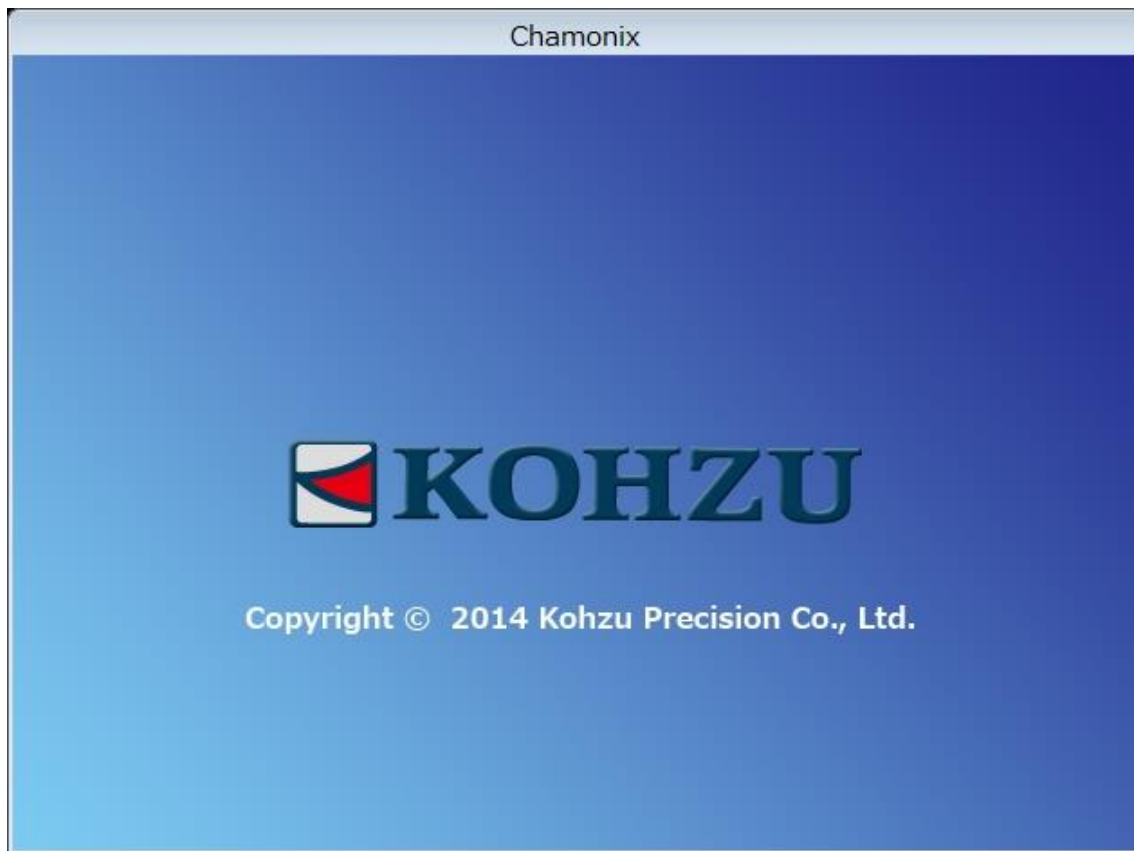


It is possible to start-up from the stored start-up file.

Information like a port number and baud rate is stored in the start-up file.

The extension of the stored start-up file is .cmxp.

5.4. Start-up Window



The screen similar to the one above should be displayed during start-up.

By default, the port number is COM1 and baud rate is 115200baud for installation.

If the start-up is not for the first time, it starts with information that was used last time.

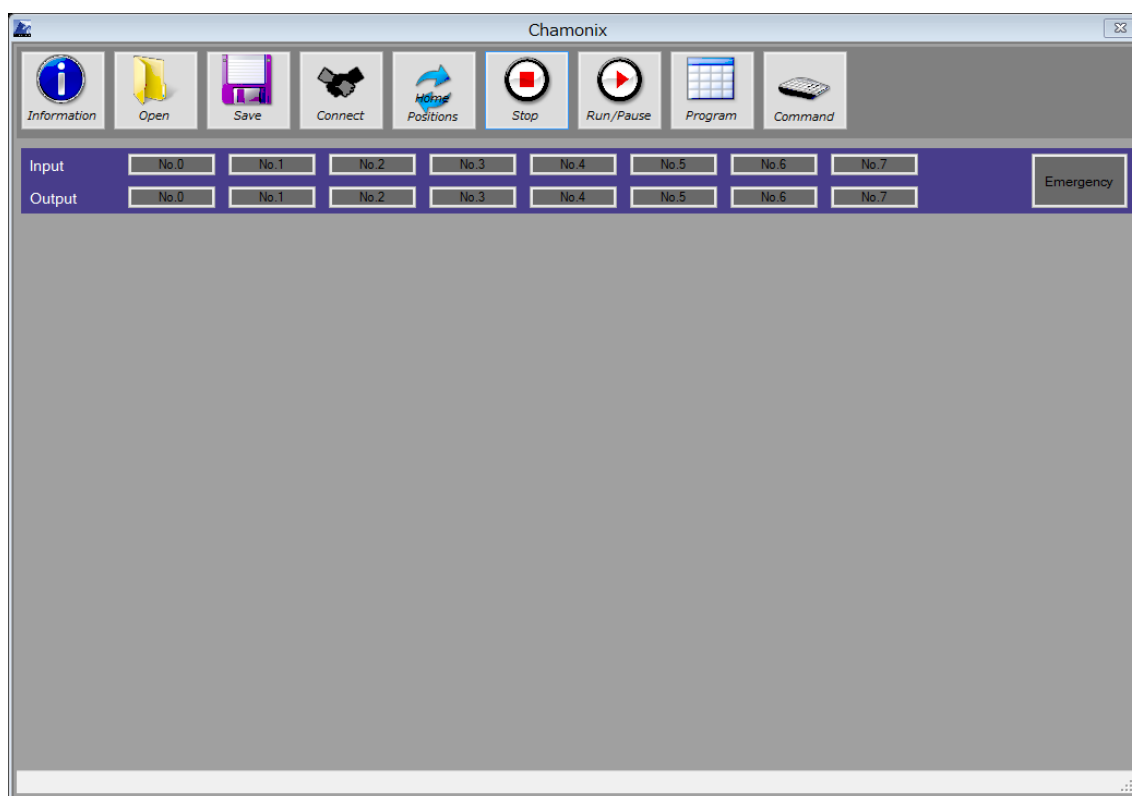
This value is stored in the Config file in C:\Program Files (x86)⁹\Common Files\KOHZU\Chamonix¹⁰.

⁹ For a 32-bit OS, the files are in C:\Program Files \Common Files\KOHZU\Chamonix.

¹⁰ Editing this file may cause a start-up failure about Chamonix.

6. Connecting the Controller Using RS-232C

6.1. Selecting COM Port



If a COM port is not selected, control display of drive axes does not appear as shown above.

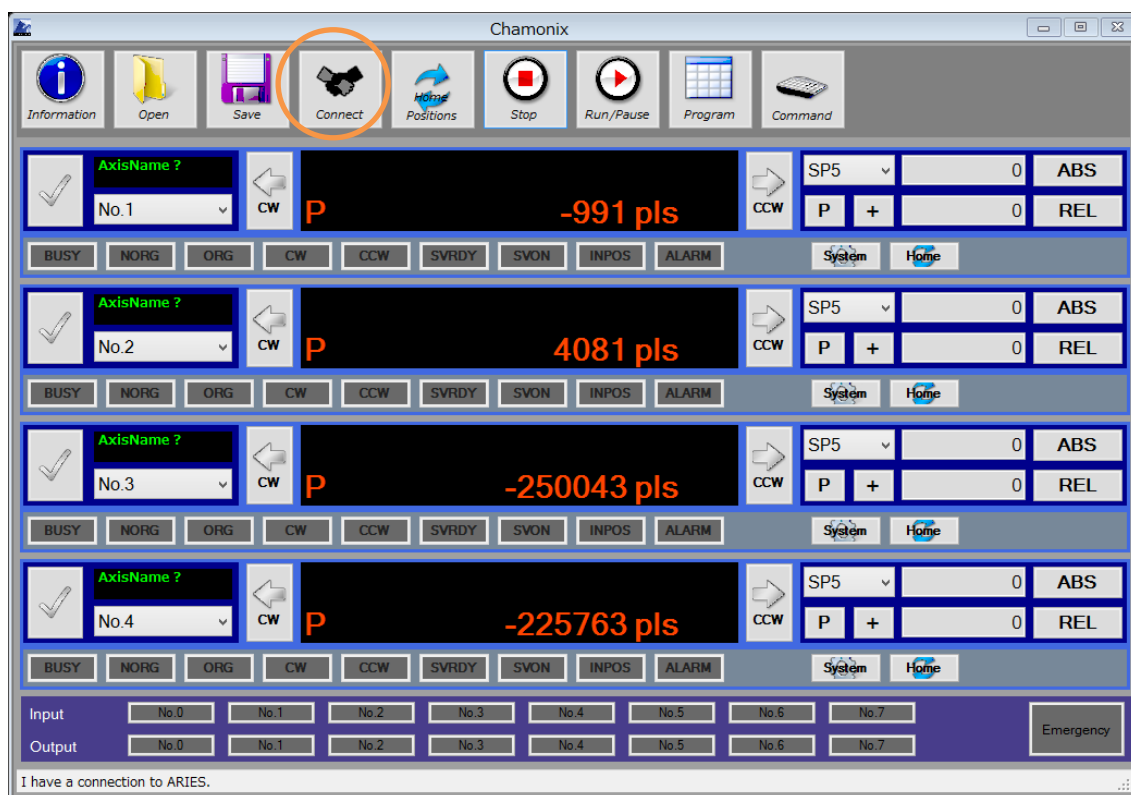
Right-click on the Connect button to display a menu and set the COM port.

The baud rate of RS-232C communication for our controller is 115200 as the factory default setting. See operation manual for each controller for more details.

Left-click on the Connect button to perform connection check and initialization.

If this application is closed after a COM port is selected and connection check is successfully complete, the next start-up is performed by using set values.

6.2. Connection Check



Press the Connect button to perform connection check and initialization.

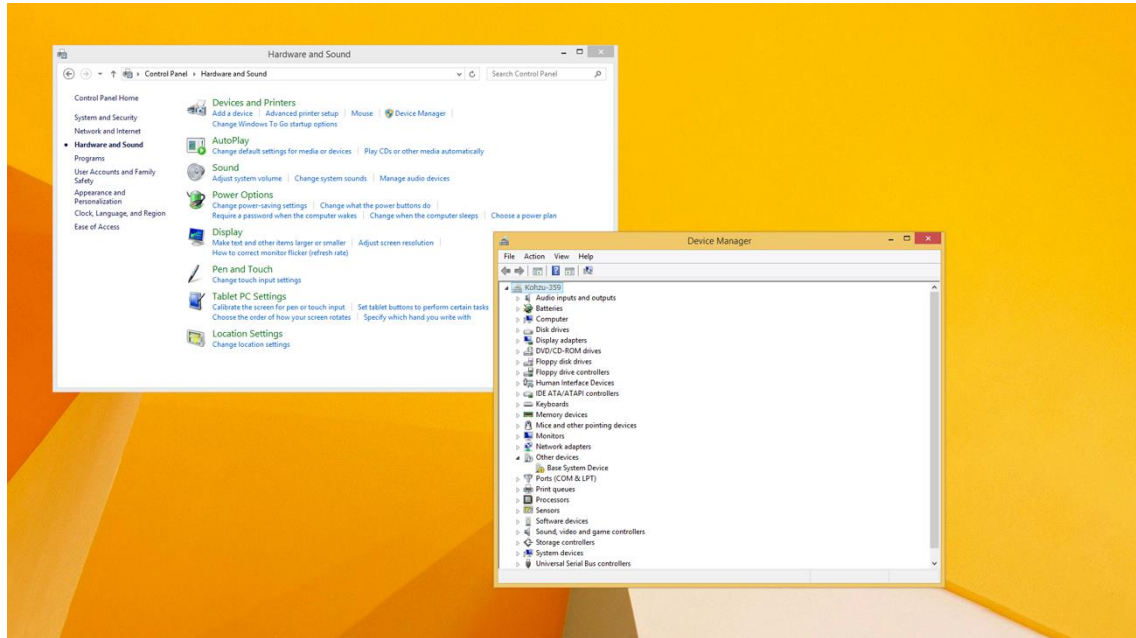
If the settings of the COM port number and baud rate are successfully read during start-up, perform a connection check and initialization automatically.

It may take five or more seconds for start-up in a case where a connection check and initialization requires time.

If the connection is established, Control displays (StageControl) of drive axes will appear as shown above. Control displays of drive axes appears up to for 4 axes¹¹.

¹¹ Like cases using CRUX or not connecting slave controller LYNX to ARIES, the displays is for 2 axes if the total number of axes is 2.

6.3. How to Check the COM PORT Number

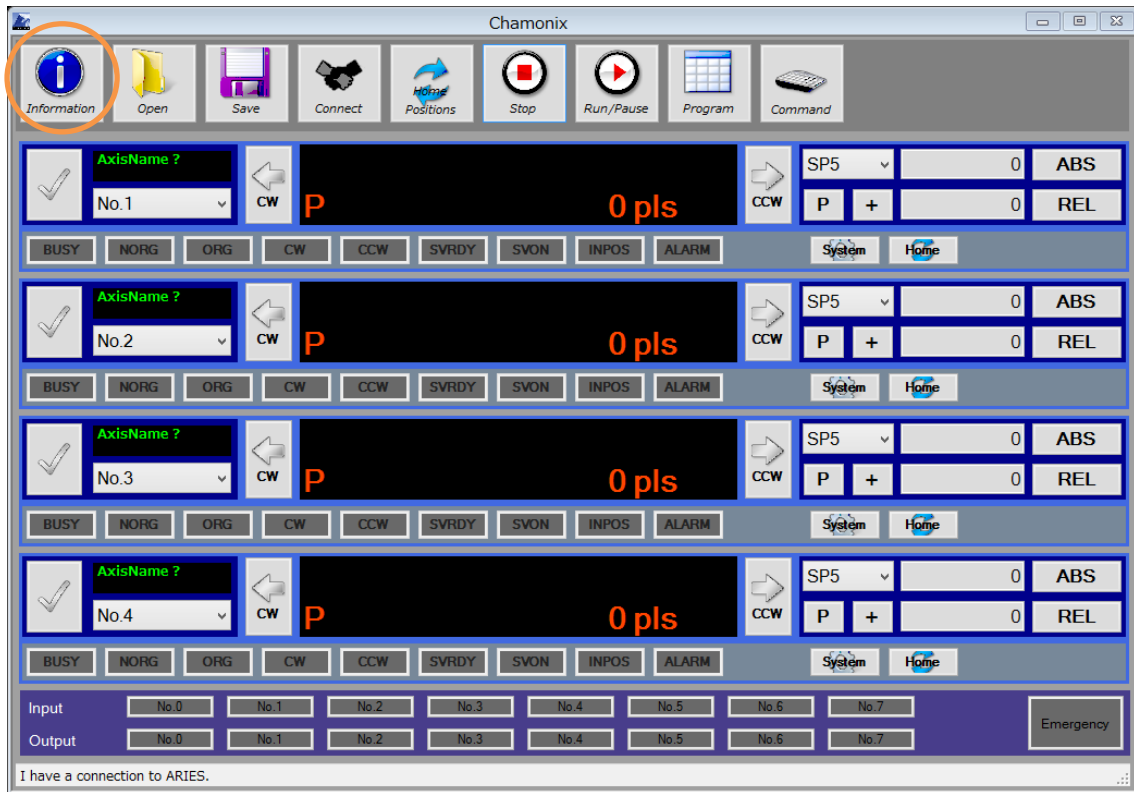


Check the port connecting controller in the Device Manager.

You can find in Control Panel > Hardware and Sound > Device Manager. To start the Device Manager, you must have administrator privileges.

7. Checking Version Information

7.1. Checking Version Information about this Application

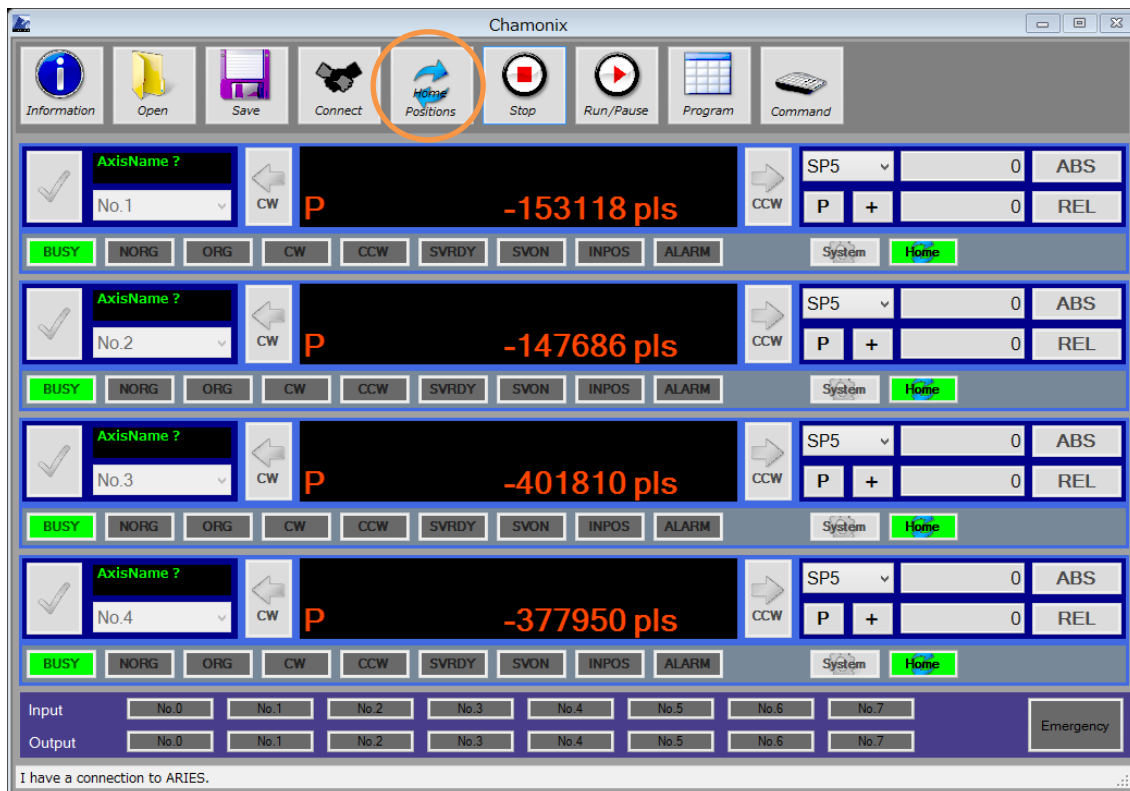


Click the Information button to check version information as shown on the next page.



8. Returning to Origin

8.1. Returning to Origin for All Axes



When you press Home Positions, all axes controlled by the controller return to origin.

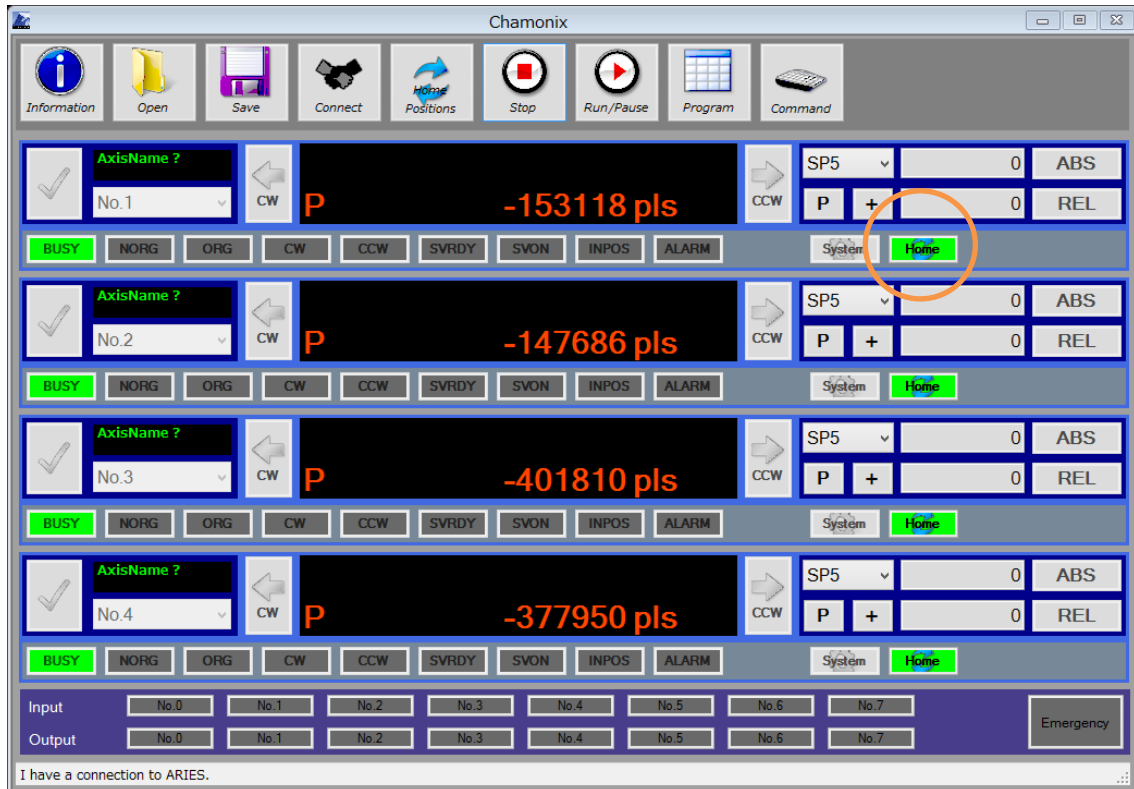
Each axis returns to origin independently without consideration for movement of other axes. Pay great care to their respective interferences.

When interference of axes could arise, return to origin for single axis as described in the next section. See a manual for your controller for patterns of returning to origin. A method to return to origin is subject to the setting value of the system parameter of the controller. Setting of system parameters is discussed in detail later.



Our company assumes no responsibility for a fault from axes interference caused by returning to origin for all axes .

8.2. Status in Returning to Origin



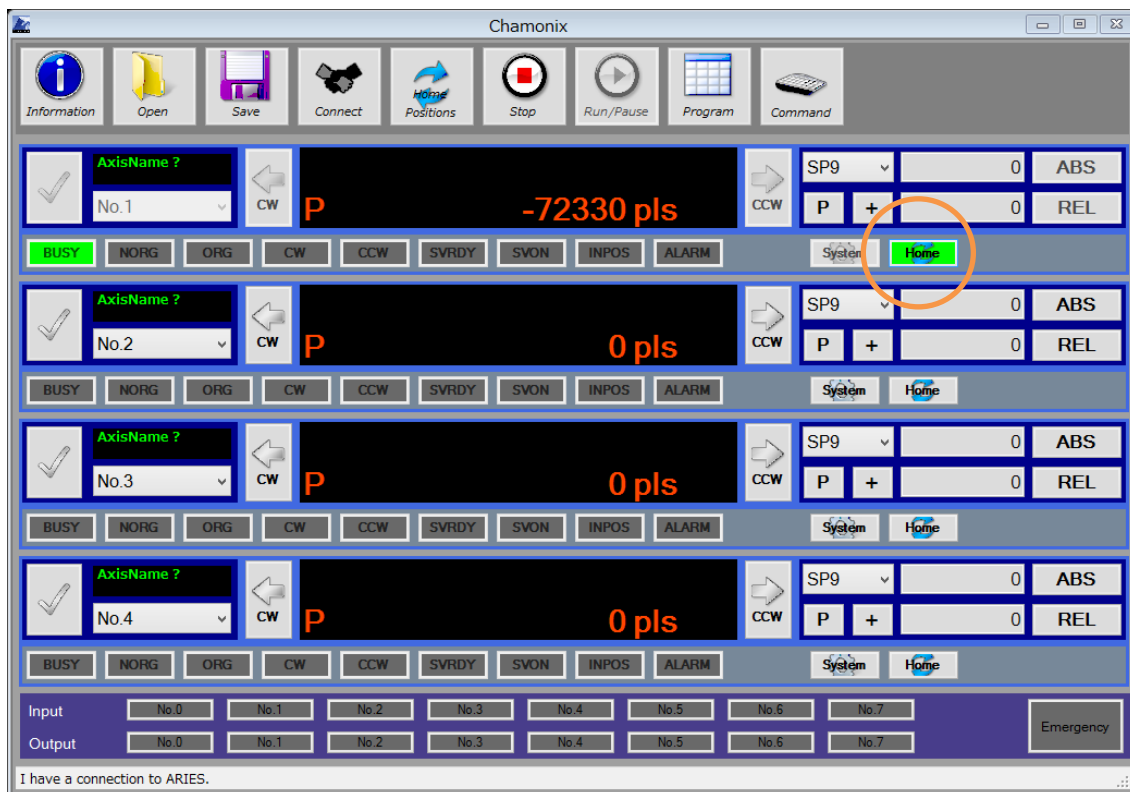
The Home button of each axis illuminates while returning to origin is performing.

The Home button goes out when returning to origin is complete.

Click a illuminating Home button to stop returning to origin for the only axis.

The BUSY indicators also illuminate during the drive.

8.3. Returning to Origin for Single Axis



You can perform retuning to origin for only a specified axis by clicking the Recovery button of each axis.

When interference may arise among axes, returning to origin for each axis one by one is recommended.

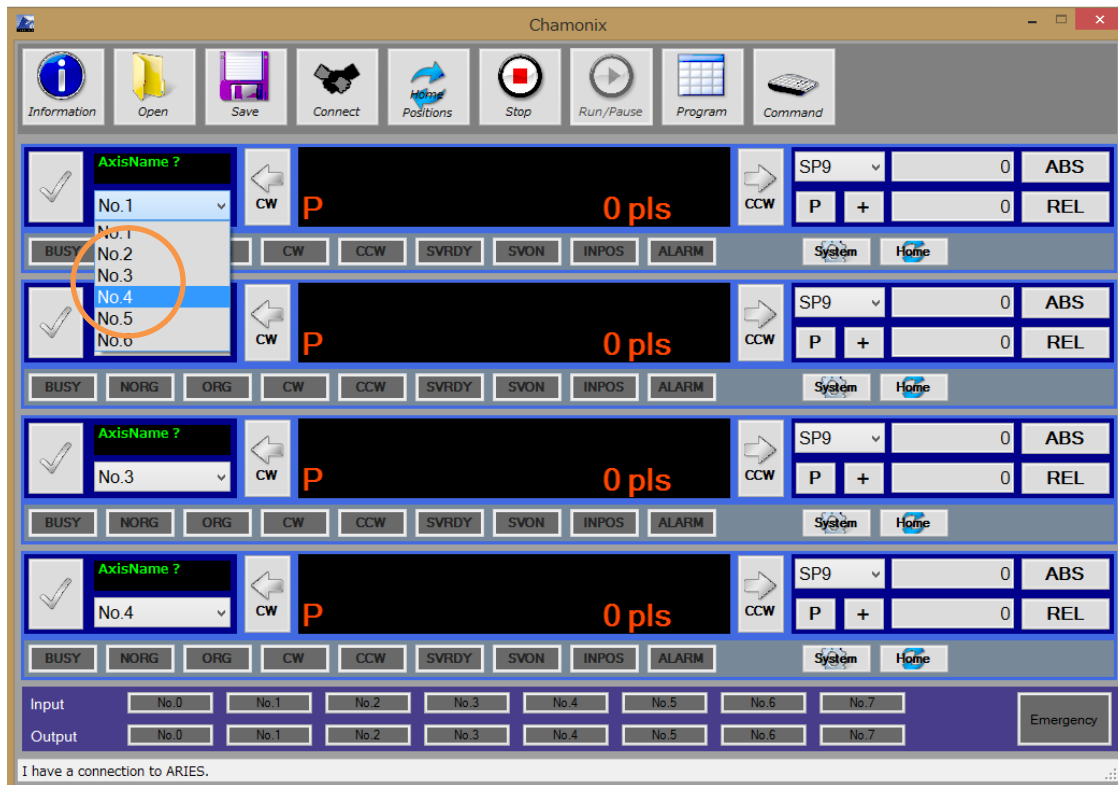
Press an illuminating button to stop.



Our company assumes no responsibility for a fault from axes interference caused by returning to origin for all axes .

9. Moving Stages

9.1. Selecting a Axis to Control

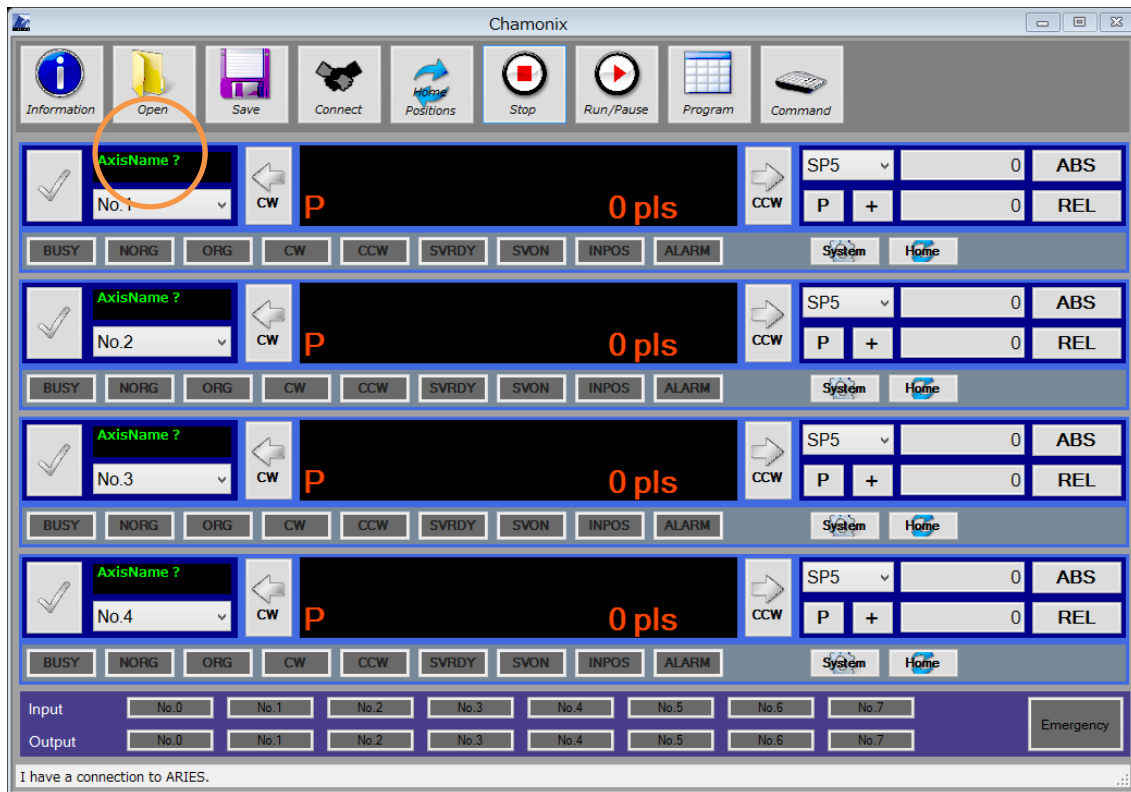


You can use a combo box to select an axis to control. You can also change an axis to control with GUI-based display during drive.

The display is up to for four axes, and the control is up to for 32 axes.

For CRUX, the control is up to for two axes. For ARIES, up to 32 axes can be controlled by adding LYNX.

9.2. Entering Axis Names

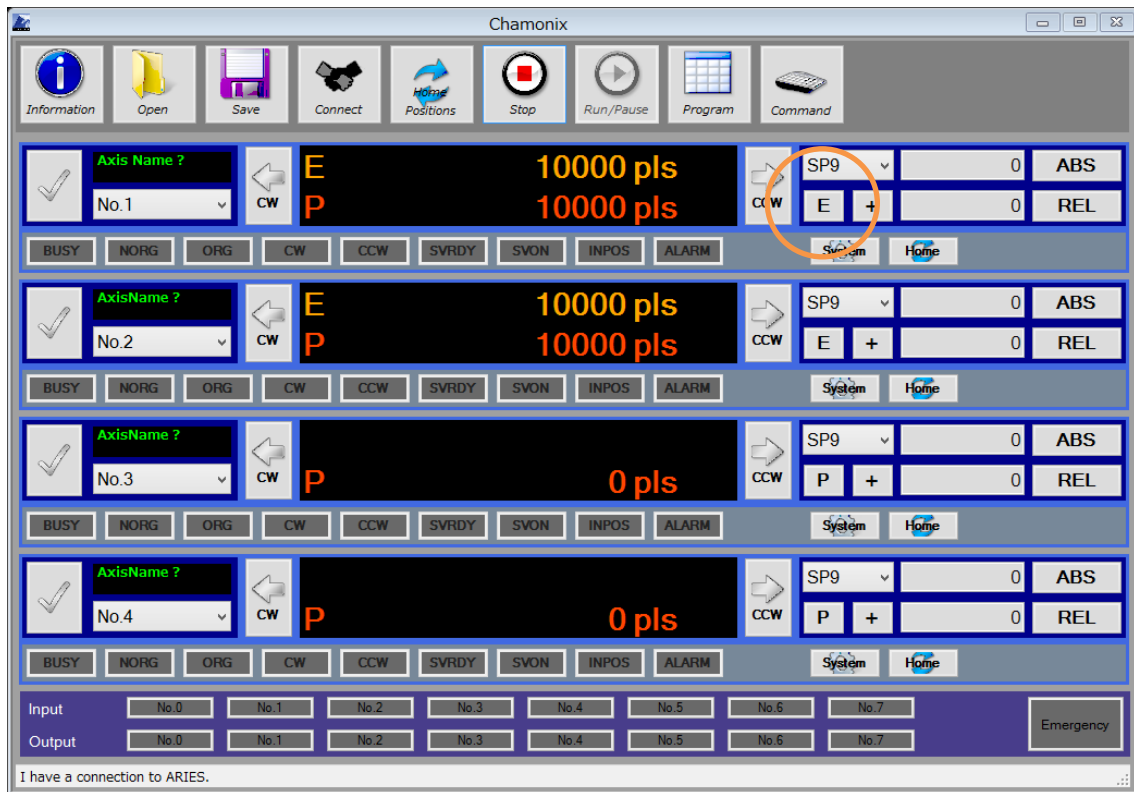


You can enter an axis name.

Axis names are saved to the start-up file or the Config file.

Language that can be entered in Windows is available. The character code is UTF-8.

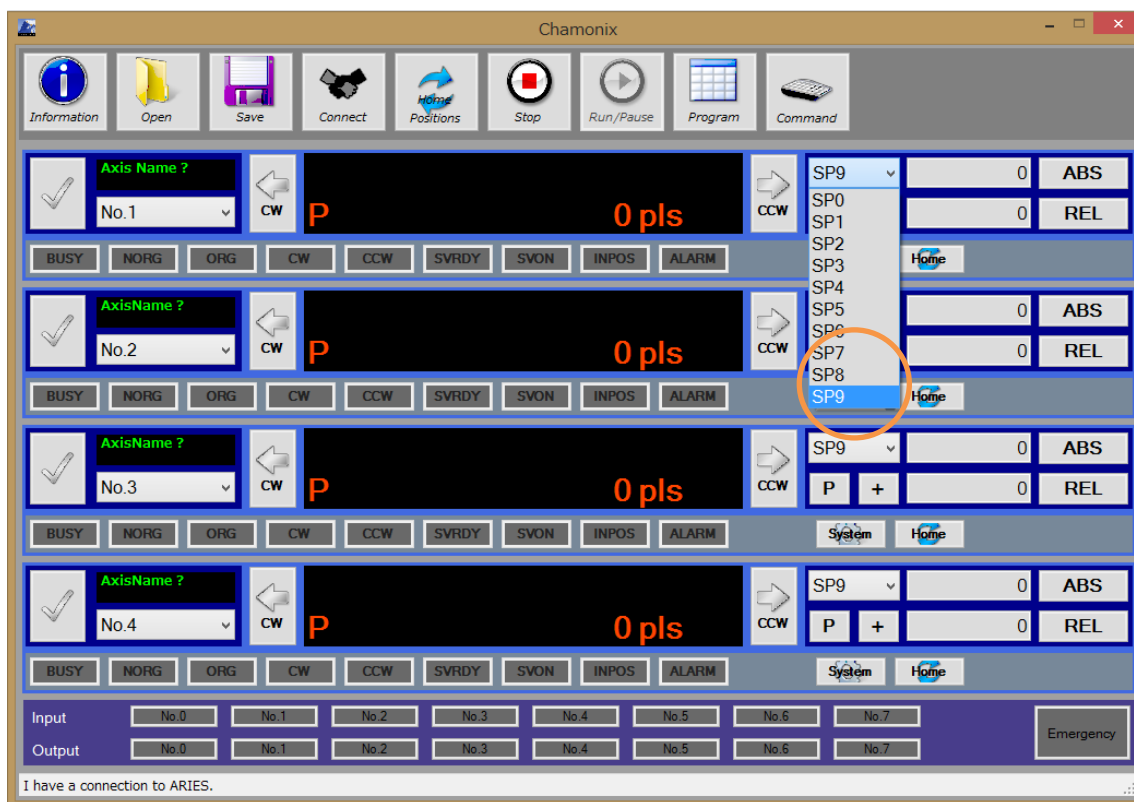
9.3. Switching the Current Position Display



Press the P button to switch to "E" and to display a current position of the encoder.

Press the E button to switch to "P" and to display a current position of the motor pulse.

9.4. Selecting Speed Table

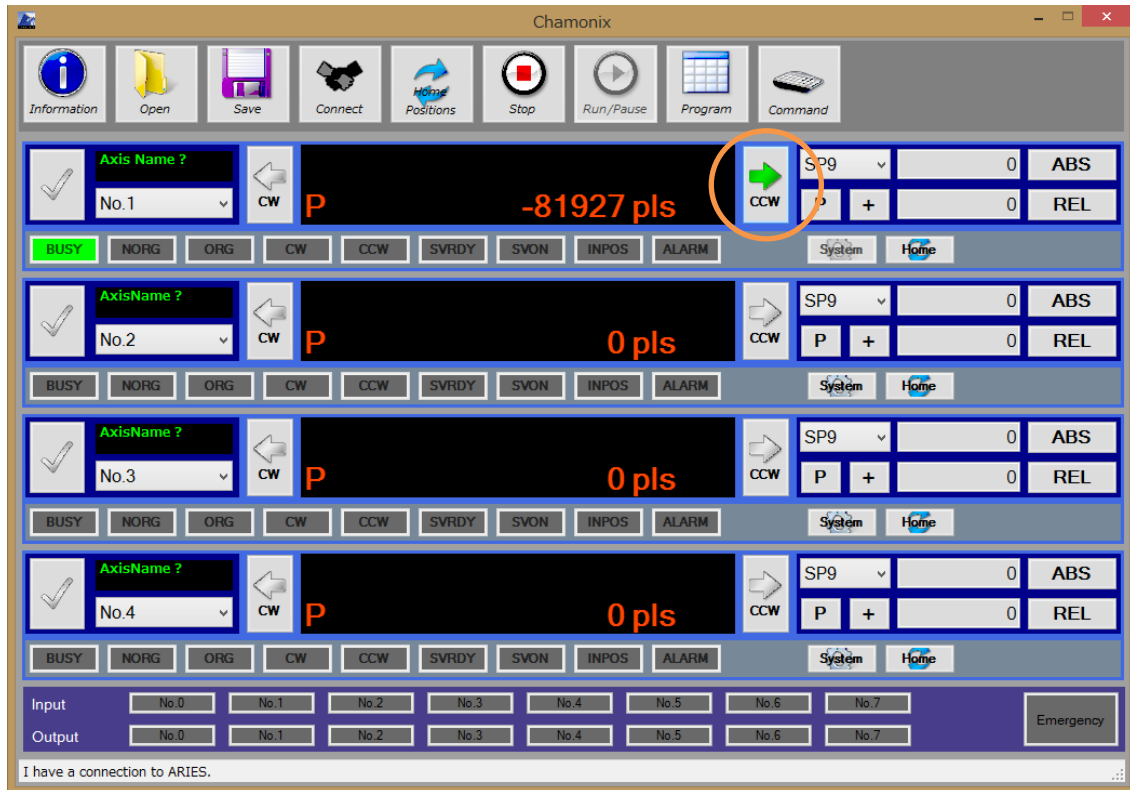


You can change the speed table of each axis from a combo box.

The setting value of a speed table can be set by changing system parameters. Changing system parameters is discussed in 12.2

Each parameter of a speed table depends on types of controllers. See operation manual for each controller for more information.

9.5. Manual Movement

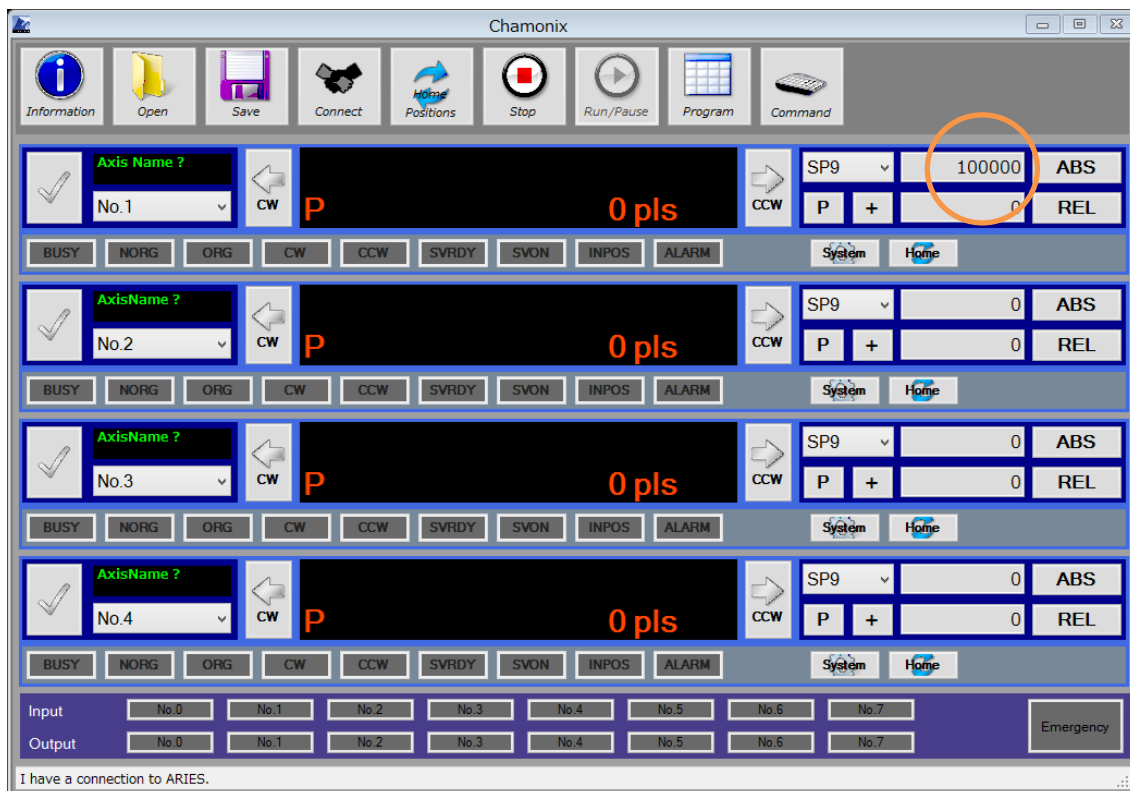


Press the CW button of each axis to move to the CW direction during the press.

Press the CCW button of each axis to move to the CCW direction during the press.

10. Deciding Position

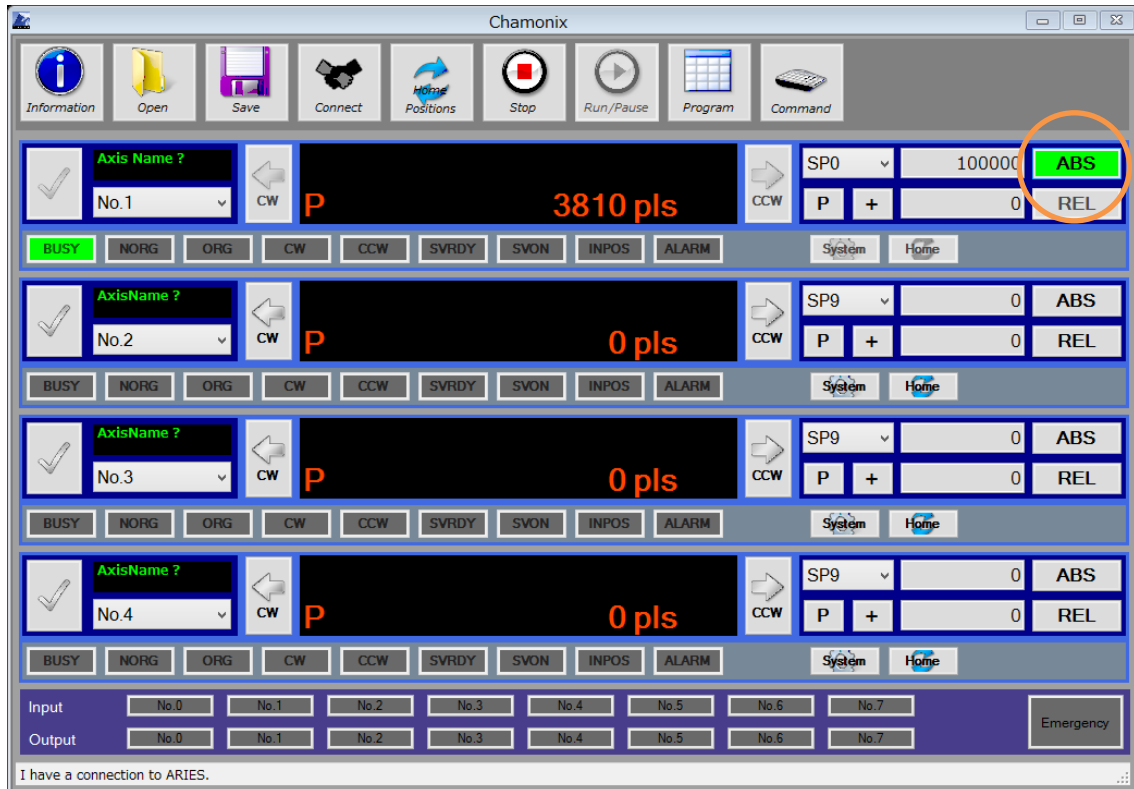
10.1. Setting of Deciding Position with Absolute Position Control



Enter an absolute position of its destination in this text box.

If a conversion value of control resolution is set in the system parameter, enter the conversion value. By default, setting for input uses motor pulse.

10.2. Executing of Deciding Position with Absolute Position Control

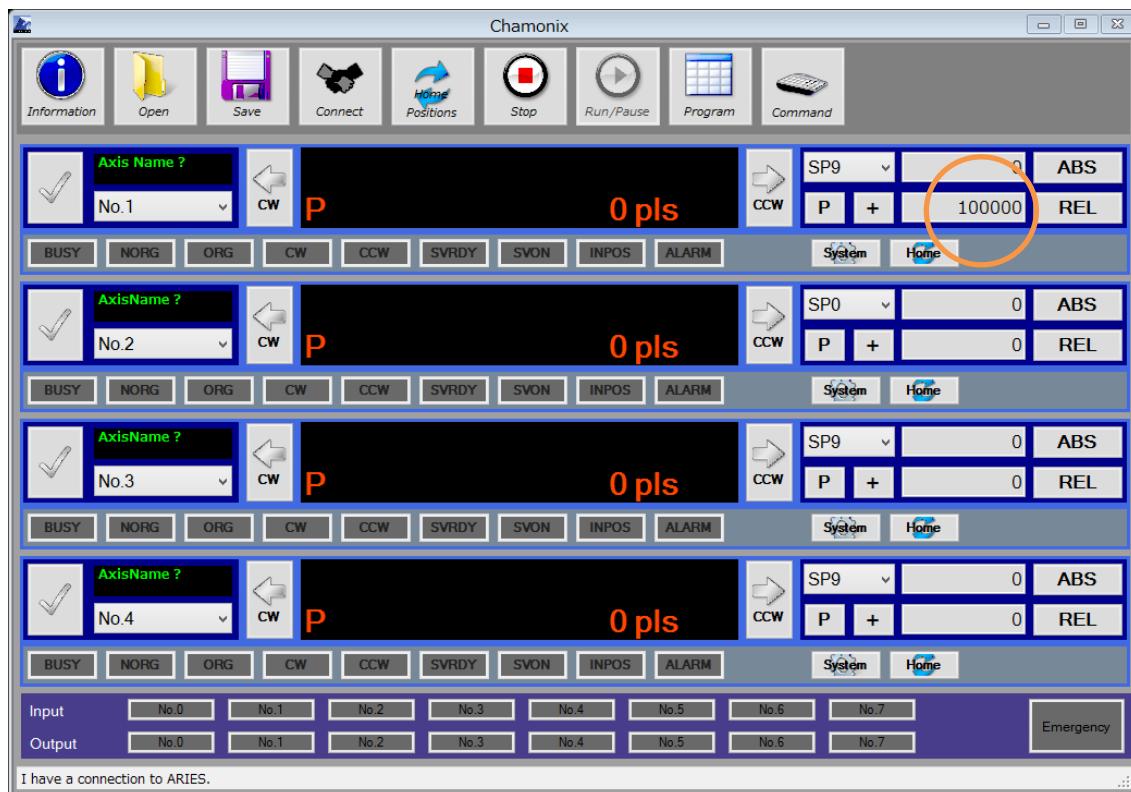


Press the ABS button with an absolute position to decide the position using absolute position control.

Press an illuminating ABS button to stop the only axis.

The BUSY indicator illuminates while the motor is driving.

10.3. Setting of Deciding Position with Relative Position Control

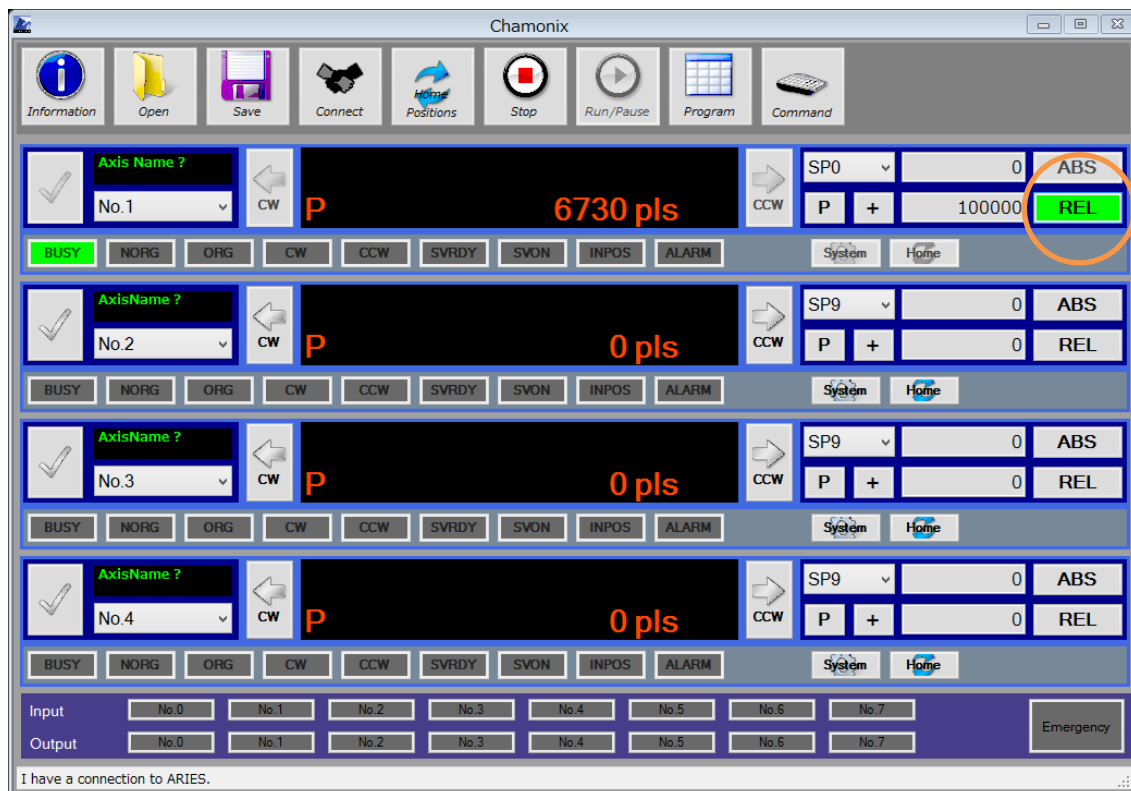


Enter a relative position of its destination in this text box.

Click a sign button to change the sign.

If a conversion value of control resolution is set in the system parameter, enter the conversion value. By default, setting for input uses motor pulse.

10.4. Executing of Deciding Position with Relative Position Control

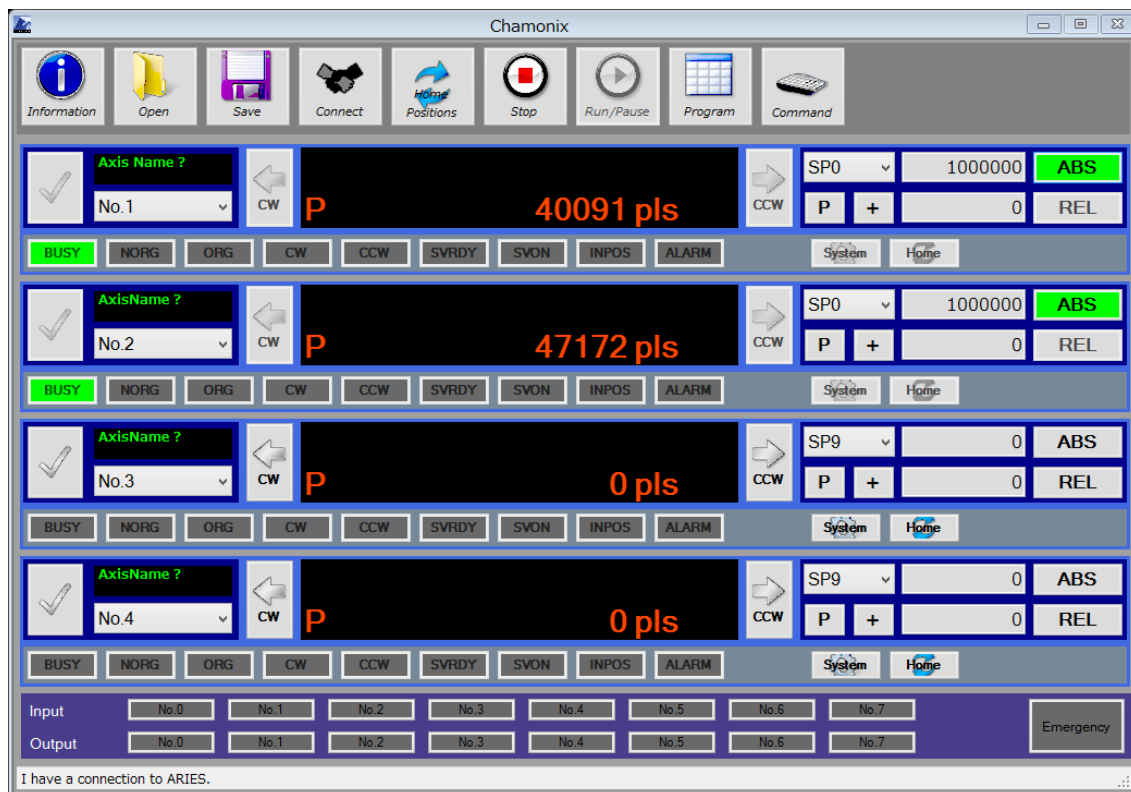


Click the REL button with a relative position to move the position using relative position control.

Press an illuminating REL button to stop the only axis.

The BUSY indicator illuminates during drive.

10.5. Executing Multi-Axis Drive



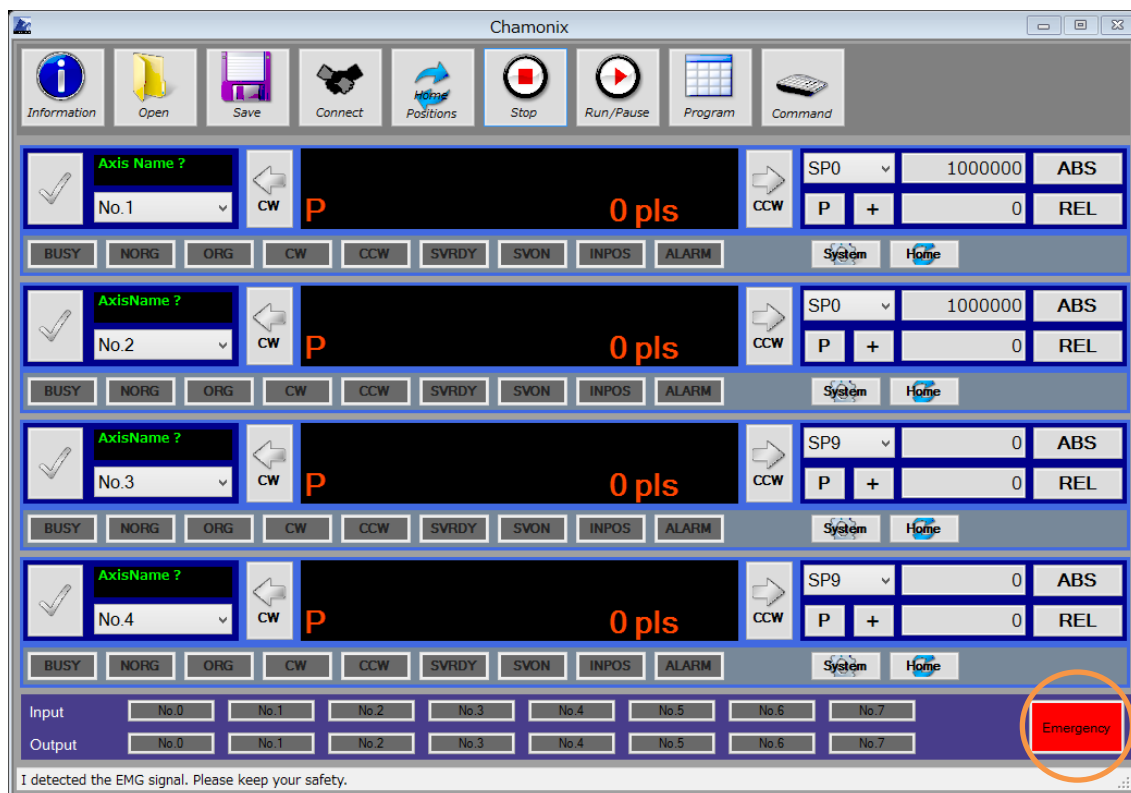
Select the check button of each axis, and press the ABS button or the REL button to drive the selected axes consecutively.

Drive instructions of APS/RPS can be transferred only to each axis in sequence at present.¹²

¹² In the future, upgrading will support various drives.

11. Receiving Emergency Stop

11.1. Receiving an Emergency Stop Signal (ARIES)



If ARIES is used for the controller, the Emergency indicator illuminates in red when an emergency stop signal is received.

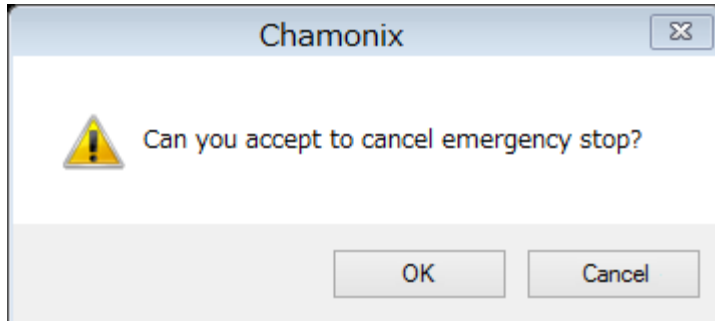
The controller will generate no motor pulse in this condition. Therefore, each motor cannot drive.

11.2. Recovering from Emergency Stop (ARIES)



For ARIES, press the Emergency button to call off the emergency stop after recovering the circuit of the emergency stop signal and checking safety.

When you press the Emergency button, the message is shown as below. Press OK to call off the emergency stop. Press Cancel to continue the emergency stop.



For recovery from an emergency stop, make sure that cause of the emergency stop is removed and safety is guaranteed.

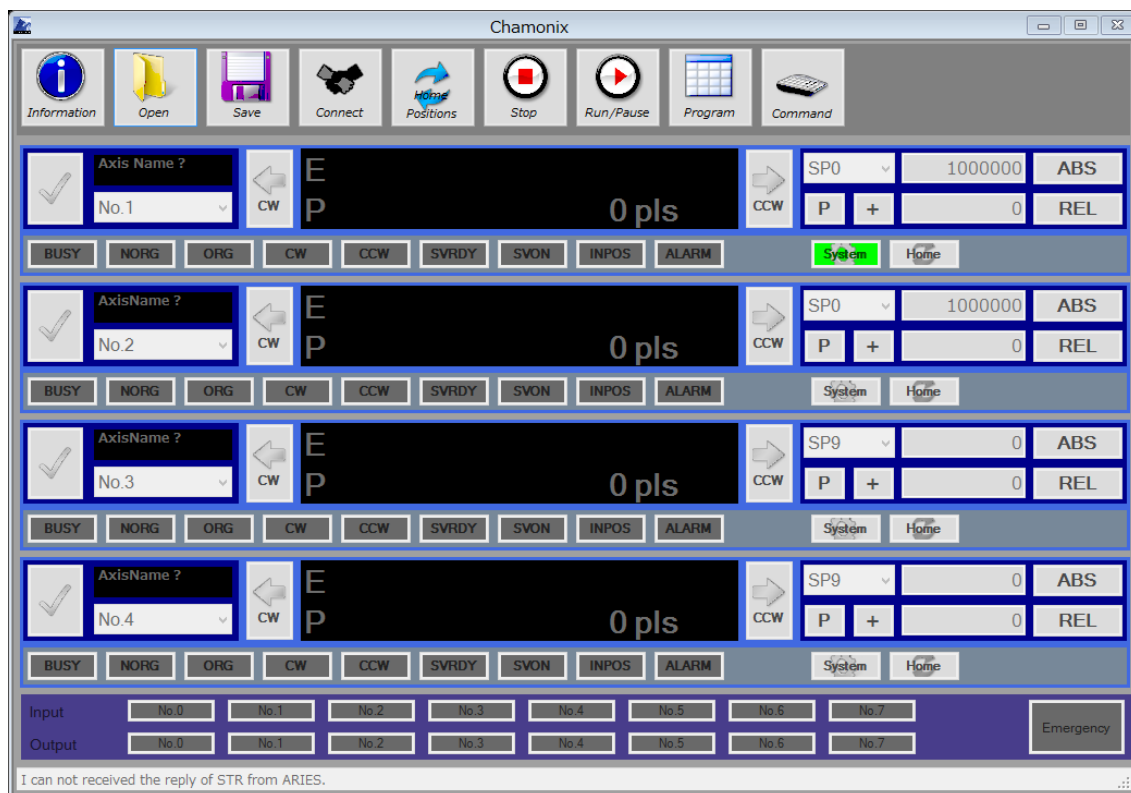
11.3. For CRUX

For CRUX, Chamonix does not display an emergency stop signal. An emergency stop works if EMG illuminates in the front panel of CRUX.

Recover the circuit with the emergency stop signal.

12. Changing System Parameters

12.1. Starting Parameter Setting



Click the System button of each drive axis to start System Parameters, a window to set system parameters. System parameters are received from the controller during start. The system parameter of each axis opens as a property shown in the next section.

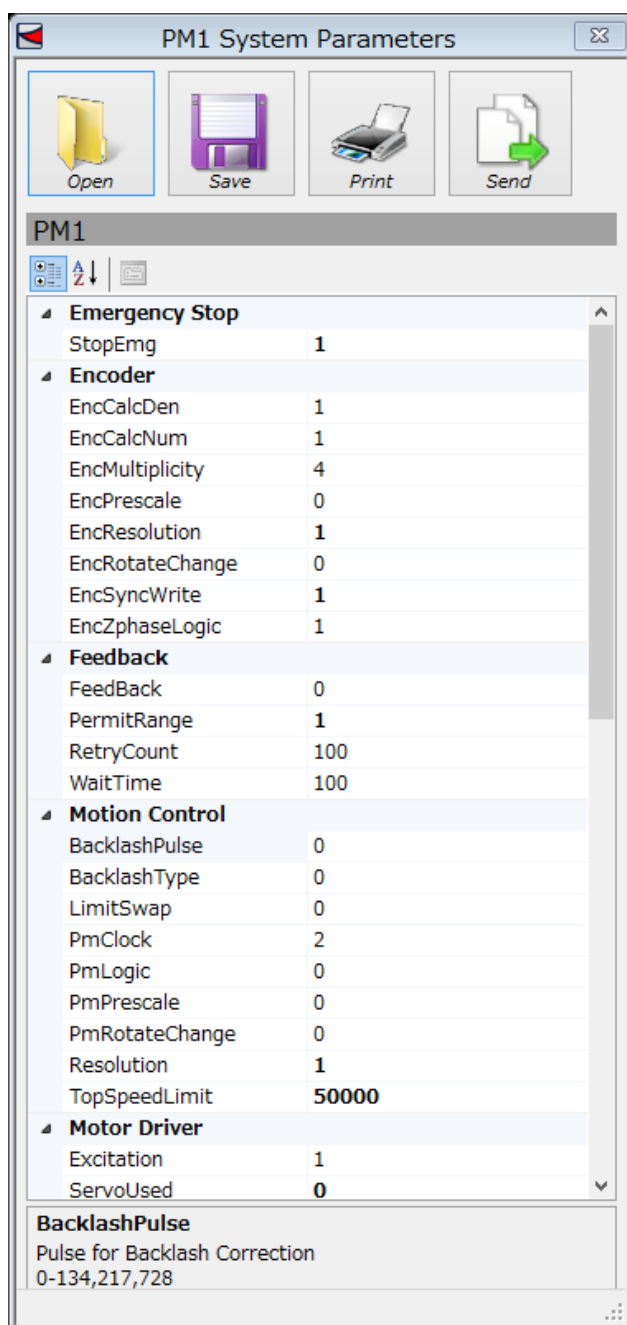
Parameter that can be set are a system parameter and speed table. These parameters are designed so that compatibility between controllers can be kept as much as possible¹³.

Features missed in a connected controller are not shown in the list. The default value of Chamonix is saved in a stored file¹⁴.

¹³ This does not mean that compatibility between different models is entirely guaranteed.

¹⁴ When a higher-level model reads a file saved in a lower-level model, parameters that is not saved in the file starts by using the Chamonix default value.

12.2. Setting Parameters



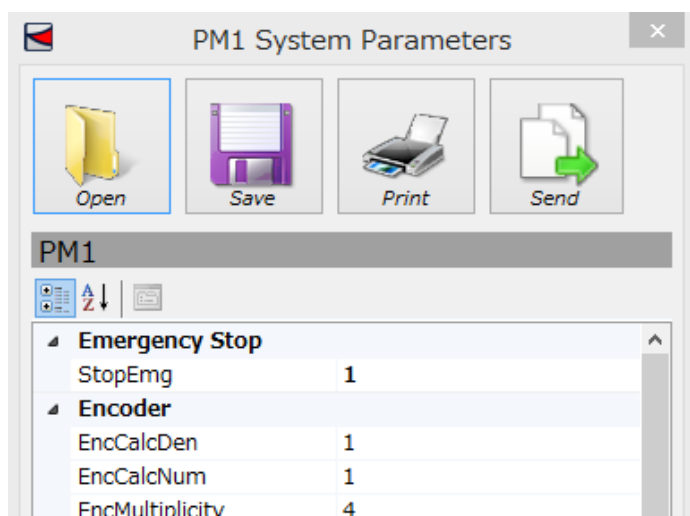
Enter values of system parameters of the controller in System Parameters to change them.

Press the Send button to reflect changed values in the controller as a whole.

Number differences by models of controllers is absorbed in integration with application's properties.

A value of property is shown as normal when a default value is set in the controller and is shown as bold when value other than a default value is set.

12.3. Saving System Parameters to Files



Press the Save button to save system parameters to files by axis.

Press the Open button in the window to read a saved file.

File formats are bin files and XML files. These files are designed for common use in SC-210/410, CRUX, and ARIES¹⁵.

Features missed in a connected controller are not shown in the list. The default value is stored in saving.

When a higher-level model reads a file saved in a lower-level model, parameters that is not saved in the file starts by using the Chamonix default value.

¹⁵ This does not mean that compatibility between different models is entirely guaranteed.

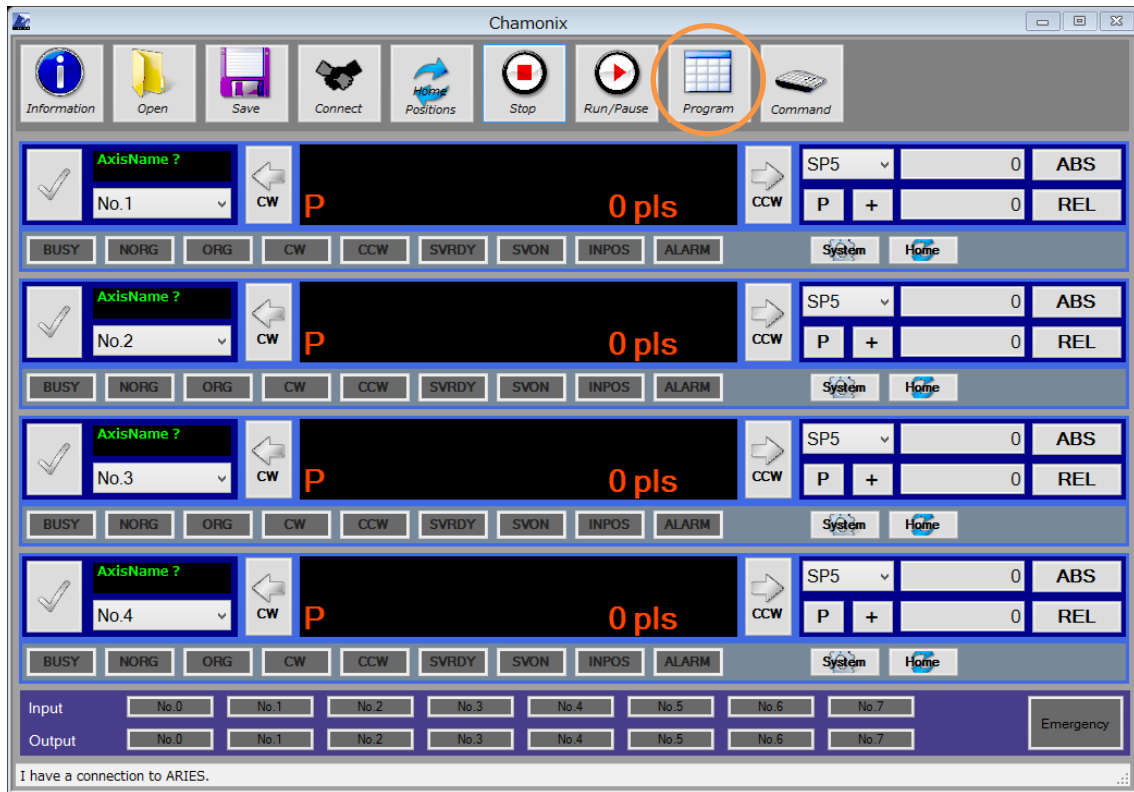
12.4. Printing Parameters

Date: 2014/07/24 20:45:37	
Controller Type: ARIES	
PM1	
Emergency Stop	
StopFmg	1
Encoder	
EncCalcDen	1
EncCalcNum	1
EncMultiplicity	1
EncPrescale	0
EncResolution	1
EncRotateChange	0
EncSyncWrite	1
EncZPhaseLogic	1
Feedback	
FeedBack	0
PermitRange	1
RotaryCount	100
WaitTime	100
Motion Control	
BacklashPulse	0
BacklashType	0
LimitsSwap	0
PmClock	2
PmLogic	0
PmPrescale	0
PmRotateChange	0
Resolution	1
TopSpeedLimit	50000
Motor Driver	
Excitation	1
ServoUsed	0
Origin Return	
OrgOffset	0
OrgScanSpeed	500
OrgType	4
Signal Logic	
LimitLogic	1
NOrigSignalLogic	0
OrgSignalLogic	0
Software Limit	
CcwSoftLimit	-∞
CwSoftLimit	∞
Speed Table	
Speed_Hi	10,8000,50,15,3
Speed_Lo	5,4000,25,10,3
SpeedNo0	500,1000,16,16,2
SpeedNo1	500,2000,20,20,2
SpeedNo2	500,3000,24,24,2
SpeedNo3	500,4000,28,28,2
SpeedNo4	500,5000,32,32,2
SpeedNo5	500,6000,36,36,2
SpeedNo6	500,7000,40,40,2
SpeedNo7	500,8000,44,44,2
SpeedNo8	500,9000,48,48,2
SpeedNo9	500,10000,52,52,2
Trigger Output	
TrgEdge	0
TrgEncPitch	1
TrgLogic	0
TrgPitch	1
TrgPulseWidth	100
TrgSource	0

Click the PRINT button to print parameters.

13. Programmable Grid

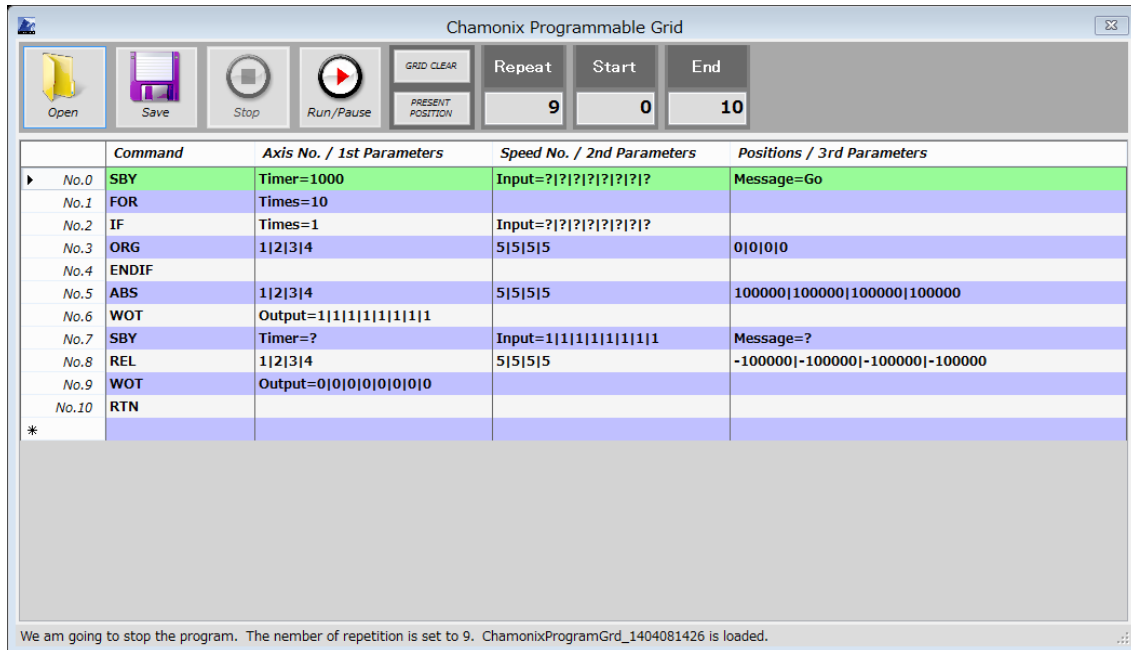
13.1. Starting Programmable Grid



Click the Program button to start Programmable Grid.

Programmable Grid is a teaching feature that made progress in conventional Setting Grid for customers and incorporated a feature like a simplified micro.

13.2. Basic Features of Programmable Grid

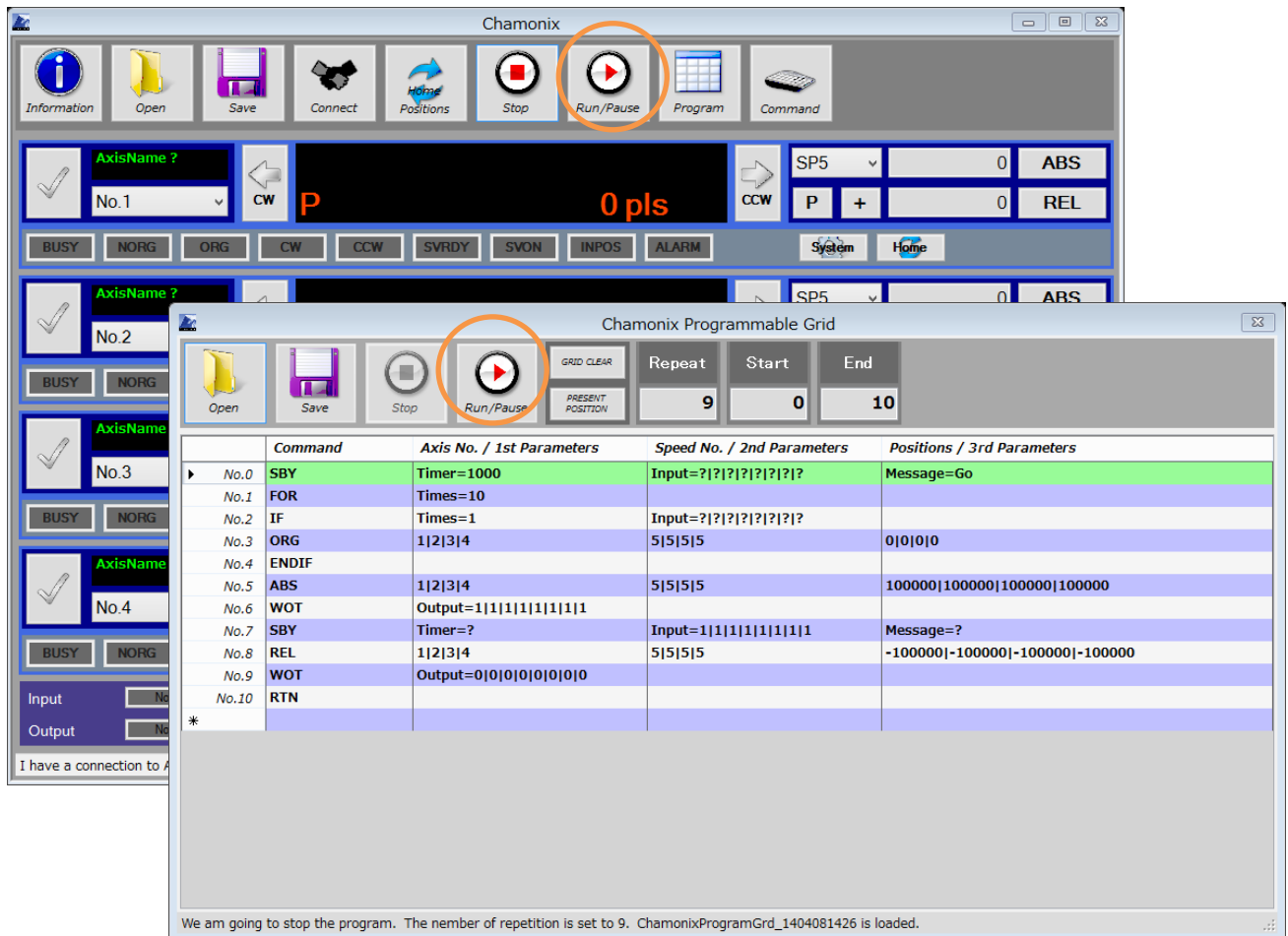


It drives automatically a stage connected with a controller according to a program written in the grid.

Commands in lines of a grid are executed in order from the top in principle.

Lines specified in Start and End are repeated the number of times specified in Repeat. If Endless or a negative number is entered in Repeat, the repeat is set to continue without end.

13.3. Running Programs

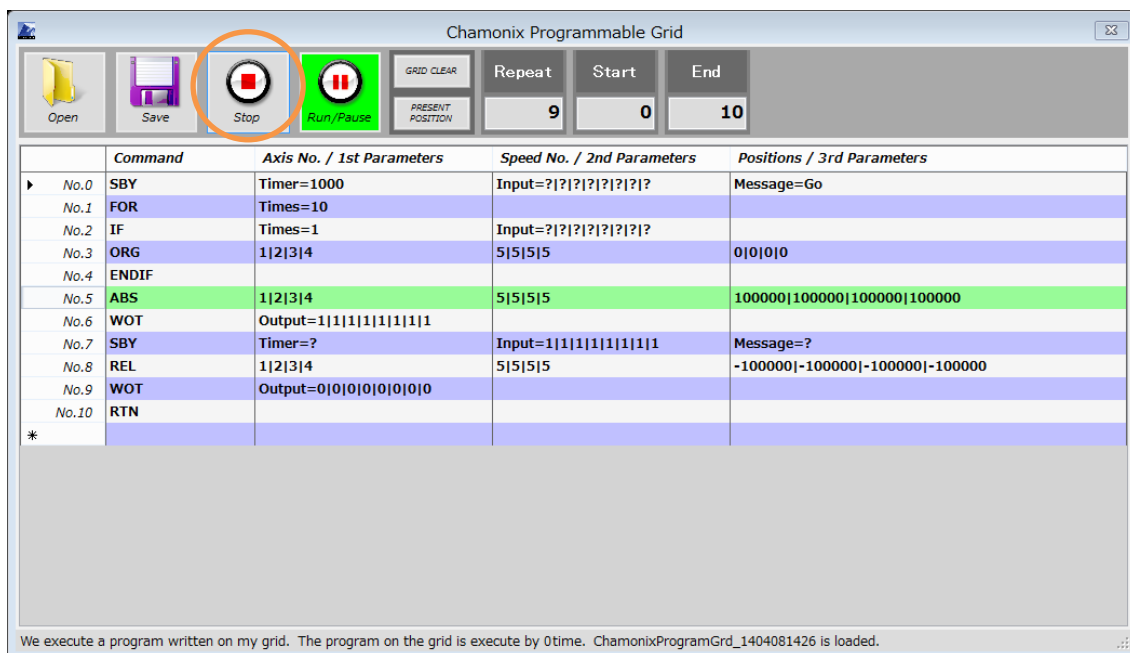


Click the RUN button in Chamonix or Chamonix Programmable Grid to run the program.

The PAUSE button replaces the RUN button and illuminates during the running of the program. The running line is colored green.

Click the PAUSE button to pause the program after finishing the running line. Press the RUN button to restart the paused program.

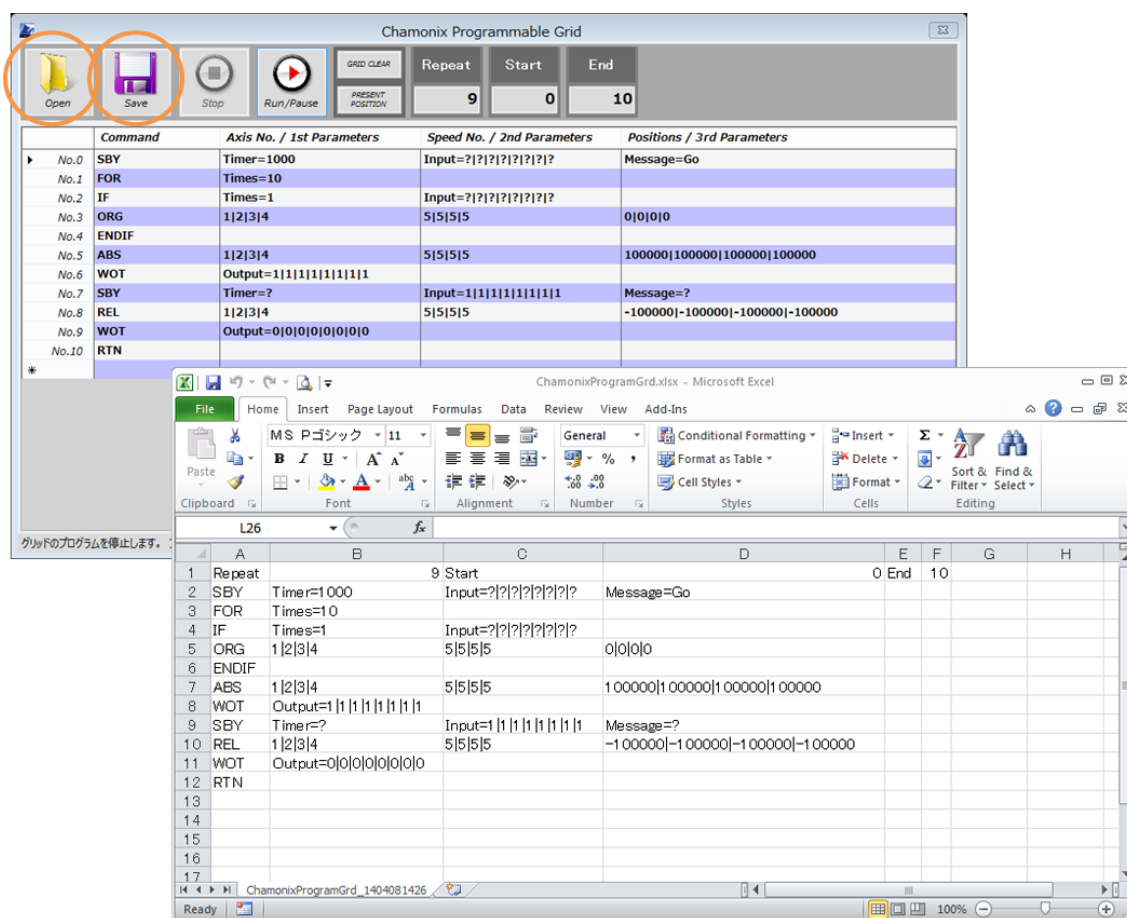
13.4. Stopping Programs



Click the STOP button to stop. Stop with the STOP button means an emergency stop, immediately stopping a program.

It is impossible to restart from the line where the program stopped. When you may restart the program, please use the PAUSE button.

13.5. Inputting and Outputting Program Data



You can save program data to files.

Data can be read from files.

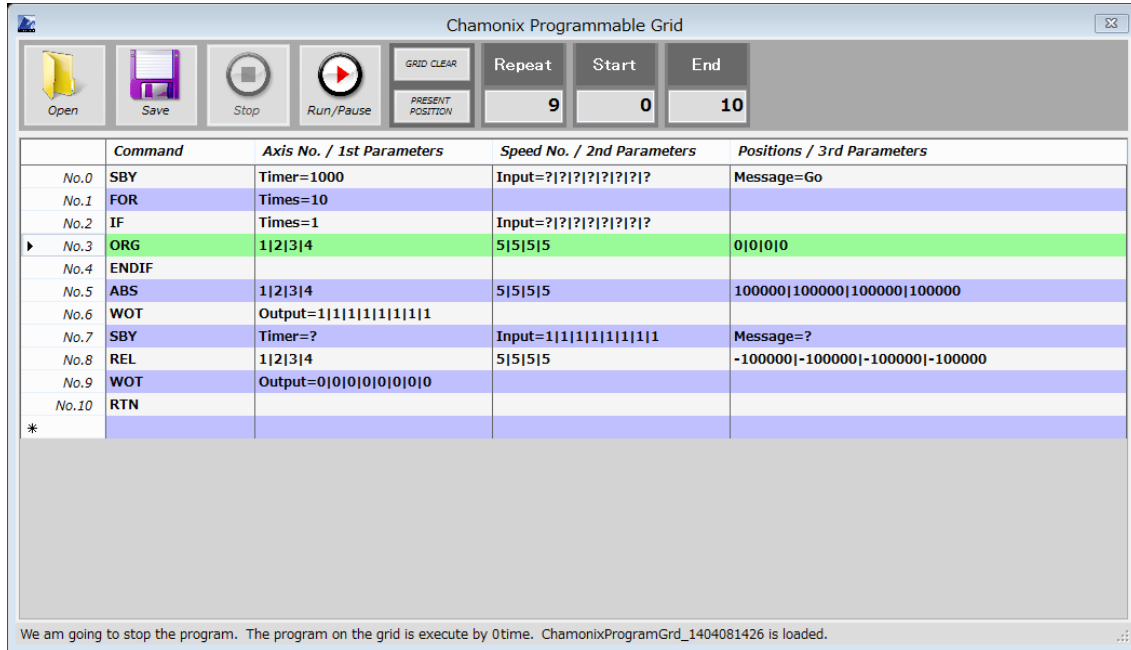
Files support DAT (standard), XML, and CSV (CSV1) formats. A DATA format is recommended.

A break character of each parameter in Parameters uses "|" or "/". It is interpreted as "|" during an internal process. This reason is the inconvenience¹⁶ in processing CSV with Microsoft Excel.

¹⁶ By default, Excel converts "1/2 " to "January 2" with a date function.

13.6. Commands of Programmable Grid

13.6.1. ORG: Returning to Origin



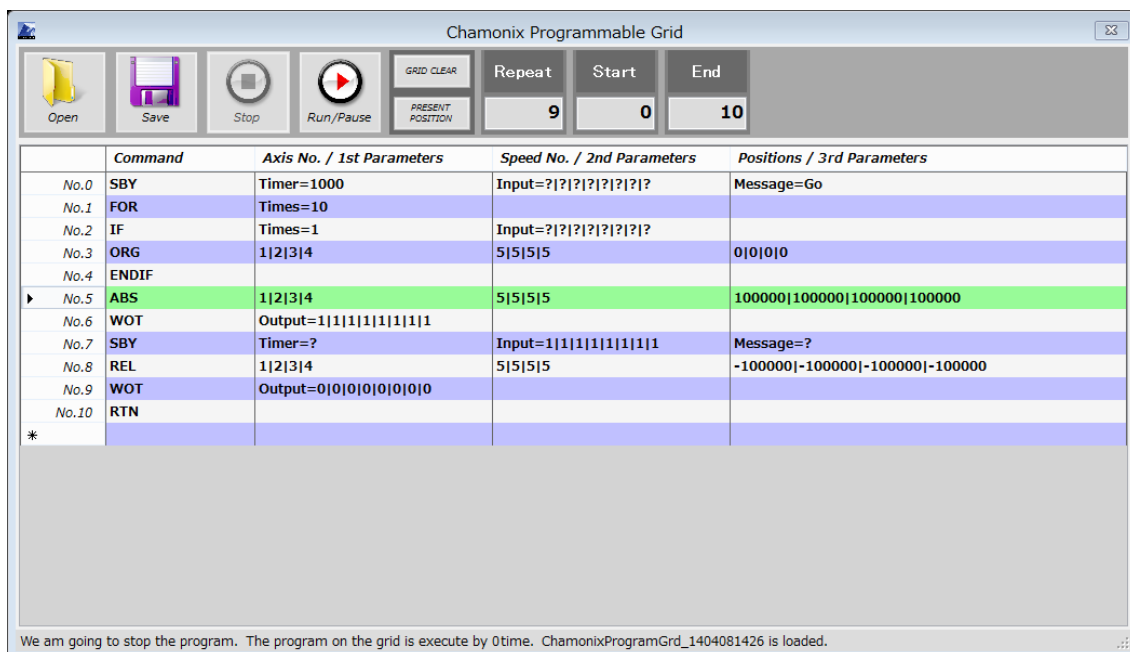
ORG performs returning to origin according to the settings of system parameters.

After completion of returning to origin,
it moves an axis specified by 1st Parameters
with a speed table specified by 2nd Parameters
to an absolute position specified by 3rd Parameters.

If the Resolution property is set in system parameters, specify a conversion value¹⁷.

¹⁷ By default, setting for input uses pulse.

13.6.2. ABS: Deciding Position with Absolute Position Control



ABS allows moving with absolute position control

to an axis specified by 1st Parameters

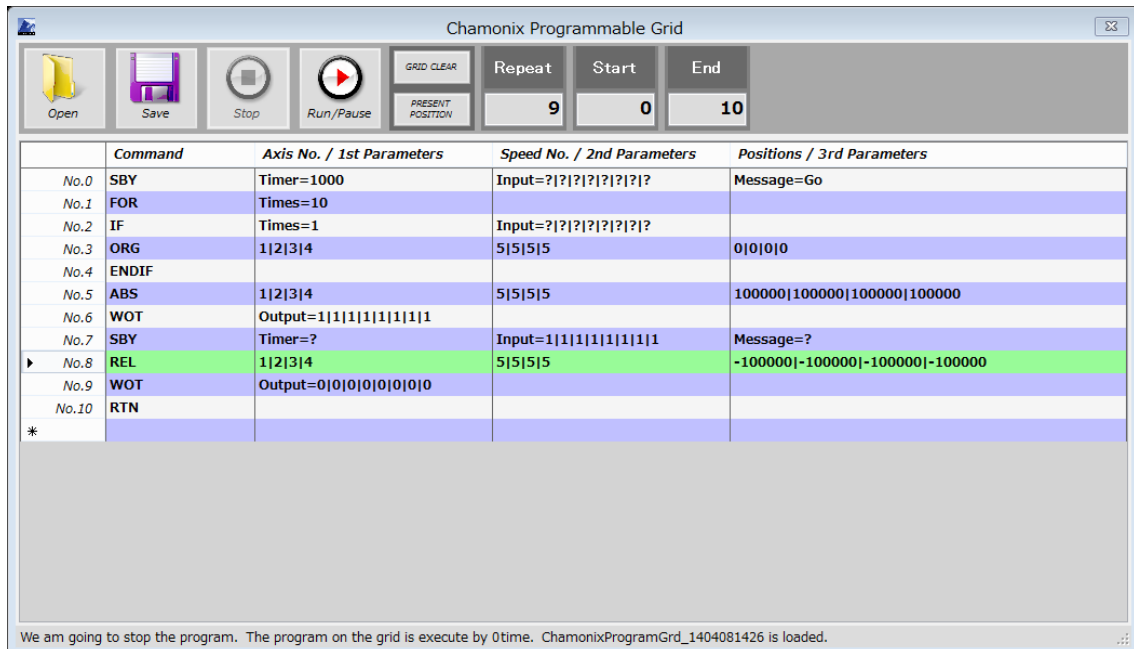
at a speed specified by 2nd Parameters

to position specified by 3rd Parameters.

If the Resolution property is set in system parameters, specify a conversion value¹⁸.

¹⁸ By default, setting for input uses pulse.

13.6.3. REL: Deciding Position with Relative Position Control



REL allows moving with relative position control

to an axis specified by 1st Parameters

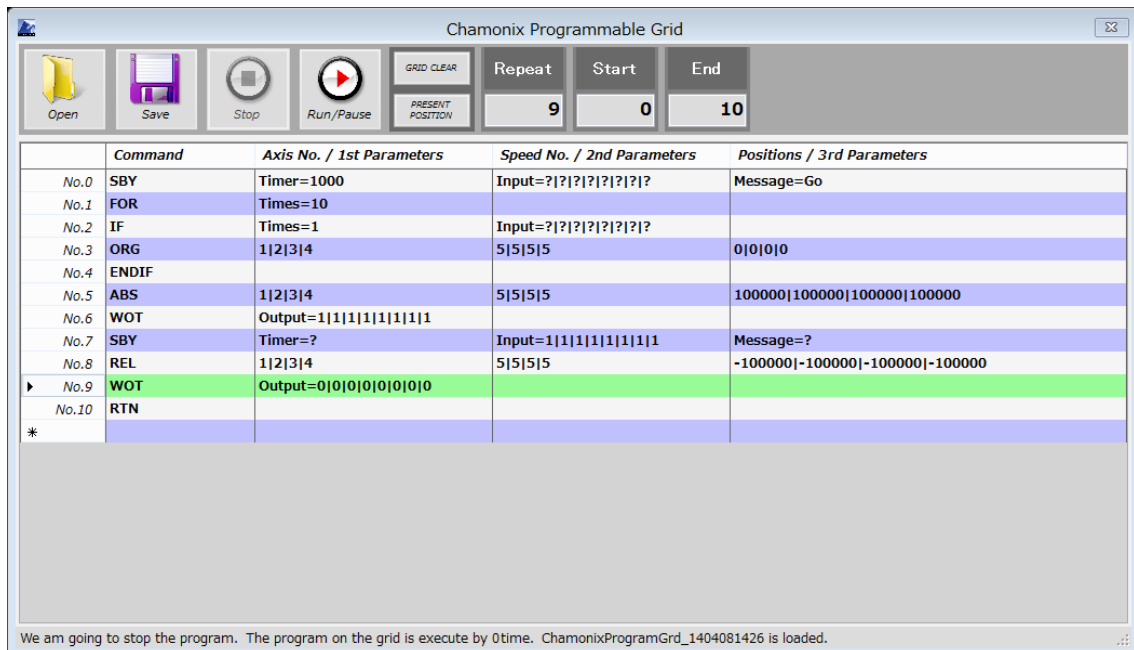
at a speed specified by 2nd Parameters

to position specified by 3rd Parameters.

If the Resolution property is set in system parameters, specify a conversion value¹⁹.

¹⁹ By default, setting for input uses pulse.

13.6.4. WOT: Controlling General-Purpose Output



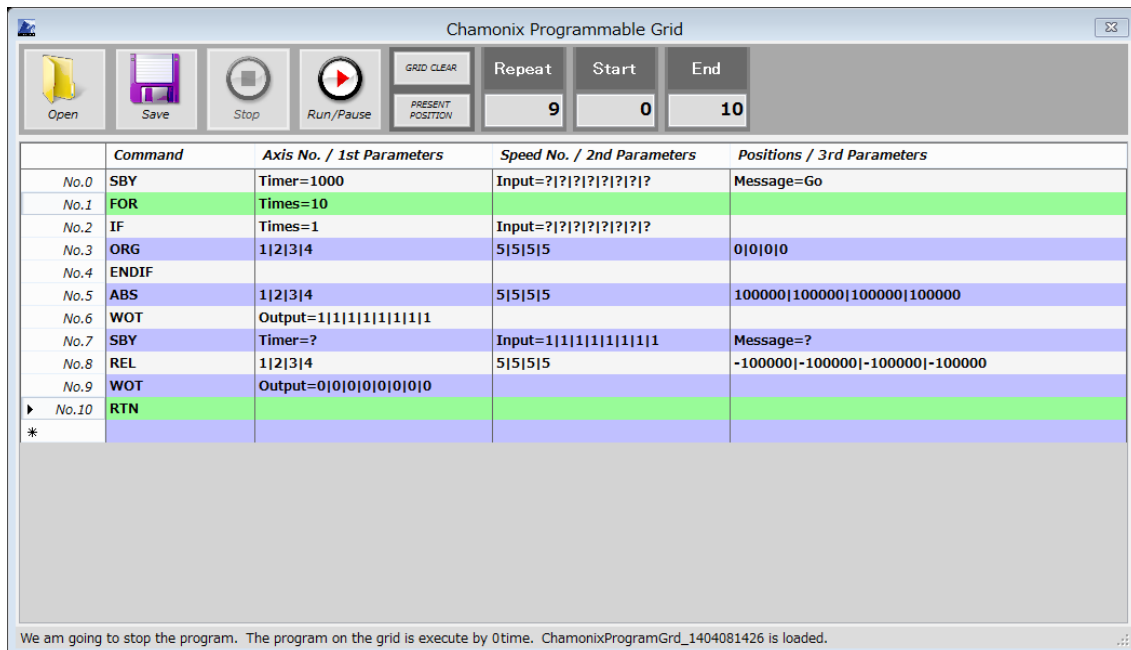
You can control the output of I/O for ARIES. The above example shows all the outputs of ARIES with 8ch are turned to ON.

Setting 1st Parameters to "Output=0|0|0|0|0|0|0|0" indicates that all outputs are turned to OFF.

The ? symbol indicates that the parameter is NULL.

When a controller that does not have I/O is connected, this command will be ignored.

13.6.5. FOR: Repeat Loop



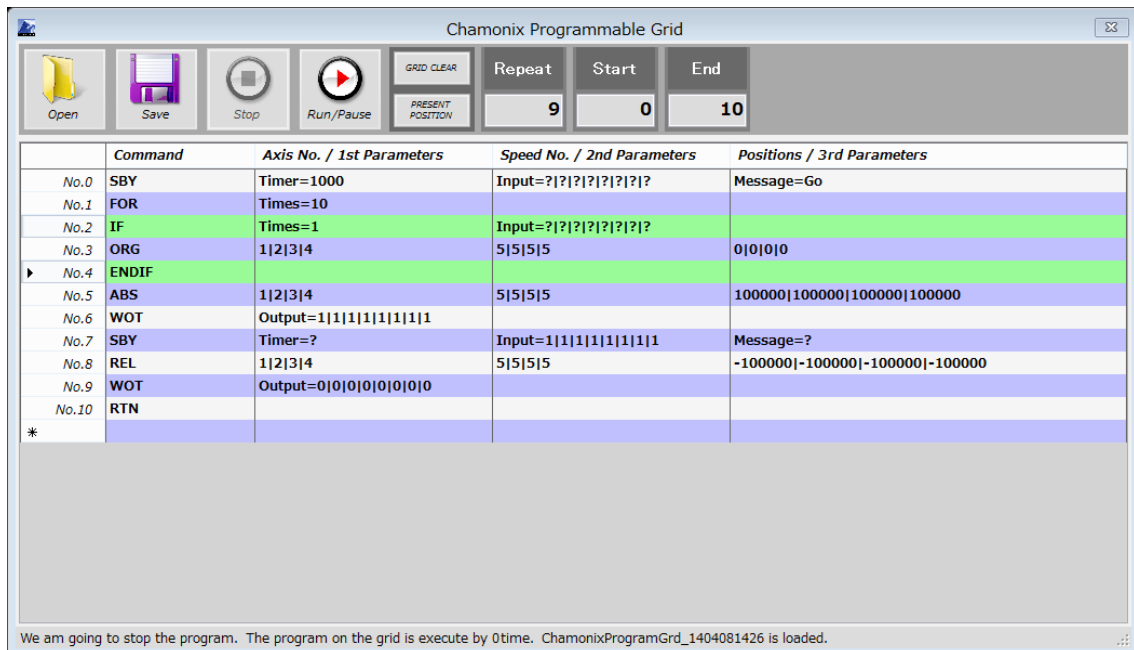
The FOR command allows repeating the number of times specified by the "Times=" parameter specifier of 1st Parameters until first RTN command in lower line than its line.

The FOR commands can nest. When the FOR commands nest, the inner loop of a FOR command runs at the beginning.

The "Times=" parameter specifier of the IF command also covers the number of times of a running loop.

If specifying the FOR command, you must insert the RTN command in the lower line than the FOR command.

13.6.6. IF: Running with conditions



When the conditions written in 1st Parameters or 2nd Parameters is met, the IF command runs lines until the ENDIF command.

You can specify the number of times of the loop by using Times= to 1st Parameters. The above example shows that the ORG command (returning to origin) is executed only if the loop is for the first time

You can specify input status of I/O for the controller as the condition to 2nd Parameters. This is ignored for a controller that does not have I/O.

If "?" is entered, the item will be ignored.

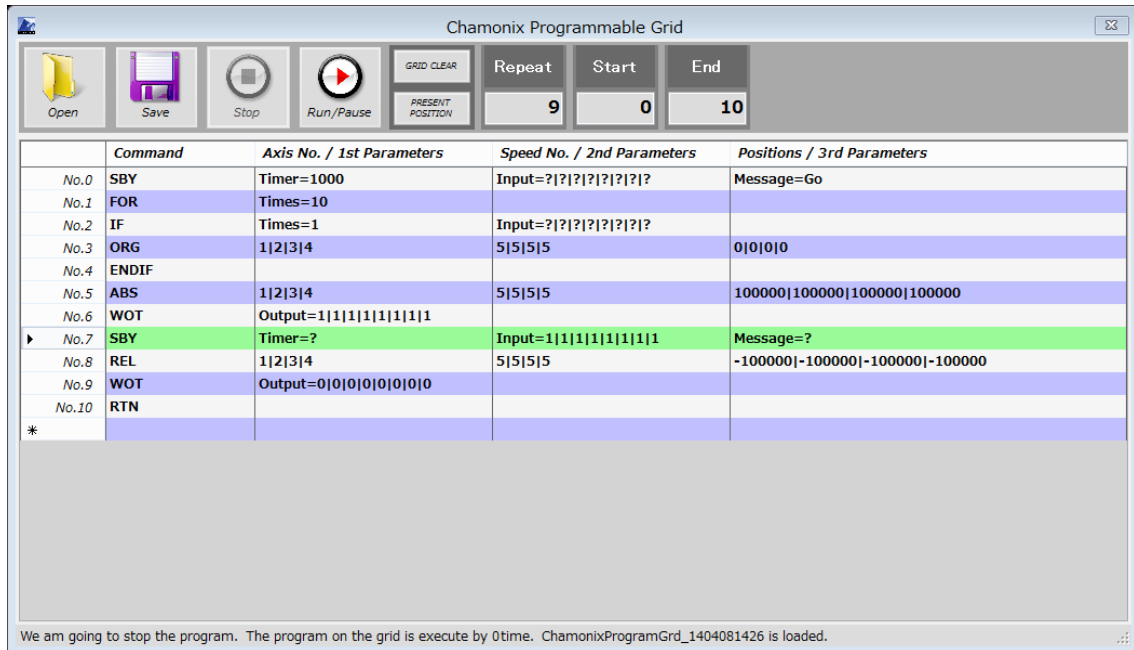
The above example shows a loop that repeats 10 times from No.1 to No.10 is set to run the No.3 line only if the loop is for the first time.

13.6.7. GOTO: Jumping to a line

The GOTO command jumps to a line specified by 1st Parameters.

Although the GOTO is convenient command, it should be noted that the program may become complicated and an incorrect program may be set.

13.6.8. SBY: Standby



The SBY command stands by until any of conditions of 1st to 3rd Parameters is met.

Parameter specifier "Time=" of 1st Parameters can specify the timer in milliseconds.

Parameter specifier "Input=" of 2nd Parameters can specify input of I/O of ARIES.

Parameter specifier "Message=" of 3rd Parameters allows going through stop when specified character strings are send to the main window of Chamonix with SendMessage of WindowsAPI.

The ? symbol indicates that a parameter is not specified.

13.6.9. Command List of Programmable Grid

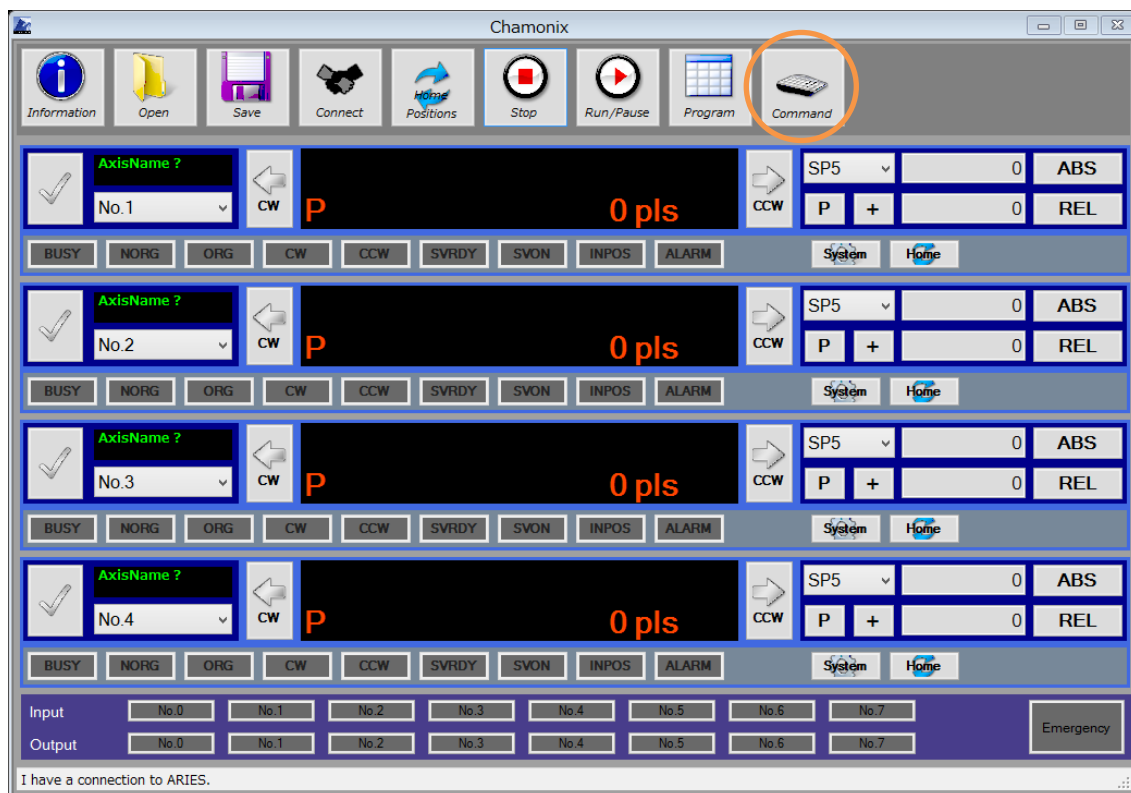
The following table lists the commands of Programmable Grid.

See descriptions of each command for more details.

ABS	•Drives with deciding position by using absolute position control
REL	•Drives with deciding position by using relative position control
ORG	•Returns to origin and decides position by using absolute position control after returning to origin is finished.
SBY	•Stands by until conditions are met.
FOR	•Repeats the number of times specified by the Times specifier until a line specified by RTN.
RTN	•Specifies a line where the FOR loop returns.
IF	•Runs lines until ENDIF if the conditions are met.
ENDIF	•Indicates the end of range where is run with conditions by the IF command.
WOT	•Sets output of I/O.
GOTO	•Jumps to specified line.

14. Entering Commands Directly

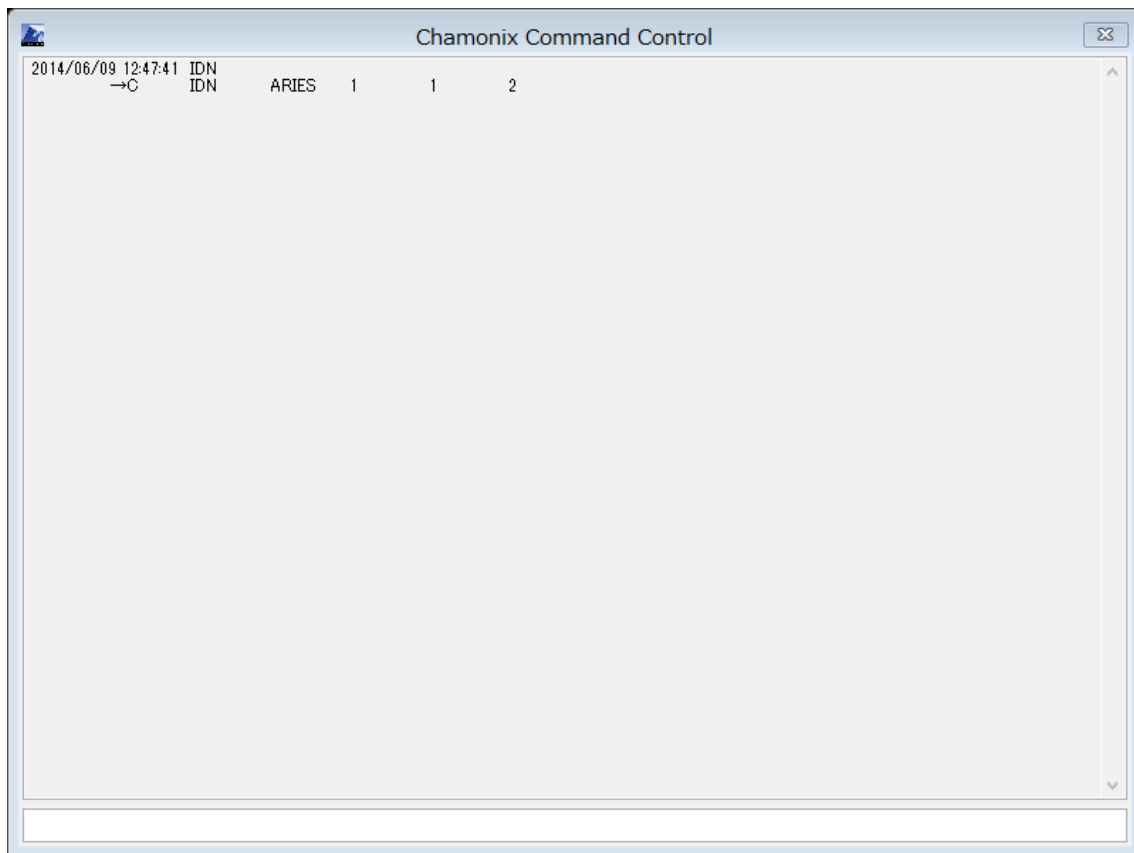
14.1. Starting Command Control



Click this button to start Chamonix Command Control.

Chamonix Command Control allows you to enter commands to send to a controller with a keyboard.

14.2. Sending Commands from Command Control



Enter a command in the send box and press Enter to send the command to the controller. The STX character²⁰ of a header and the CRLF character^{21,22} of a delimiter for RS-232 communication are added automatically. See operation manual for each controller for remote control formats. Entered lowercase letters will be converted into uppercase ones.

Time a command was sent and response from controller will appear in this window.

The application does not performed any error checking. Enter correct commands. IF an incorrect character string are sent to the controller, an error will be returned by controller's specifications. See operation manual for the controller for more details.

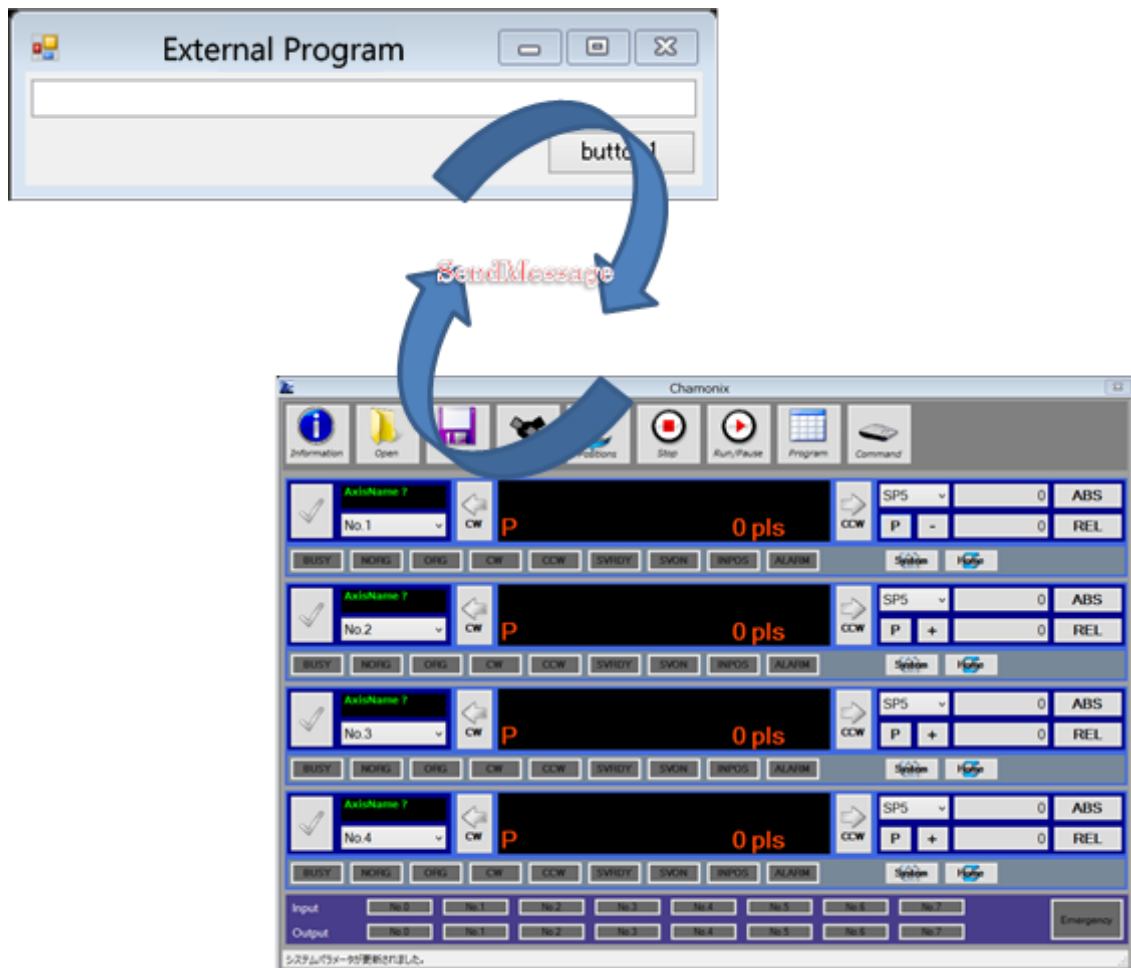
²⁰ "STX" is not three characters of "S", "T", and "X", but one control character indicating "start of text".

²¹ "CR" is not two of Roman letters, but a control character indicating return.

²² Control character "LF" indicates line feed in a similar way.

15. Control from outside using API

15.1. Control from outside with Windows API



In Chamonix, you can control drives using WM_COPYDATA of Windows API from another application.

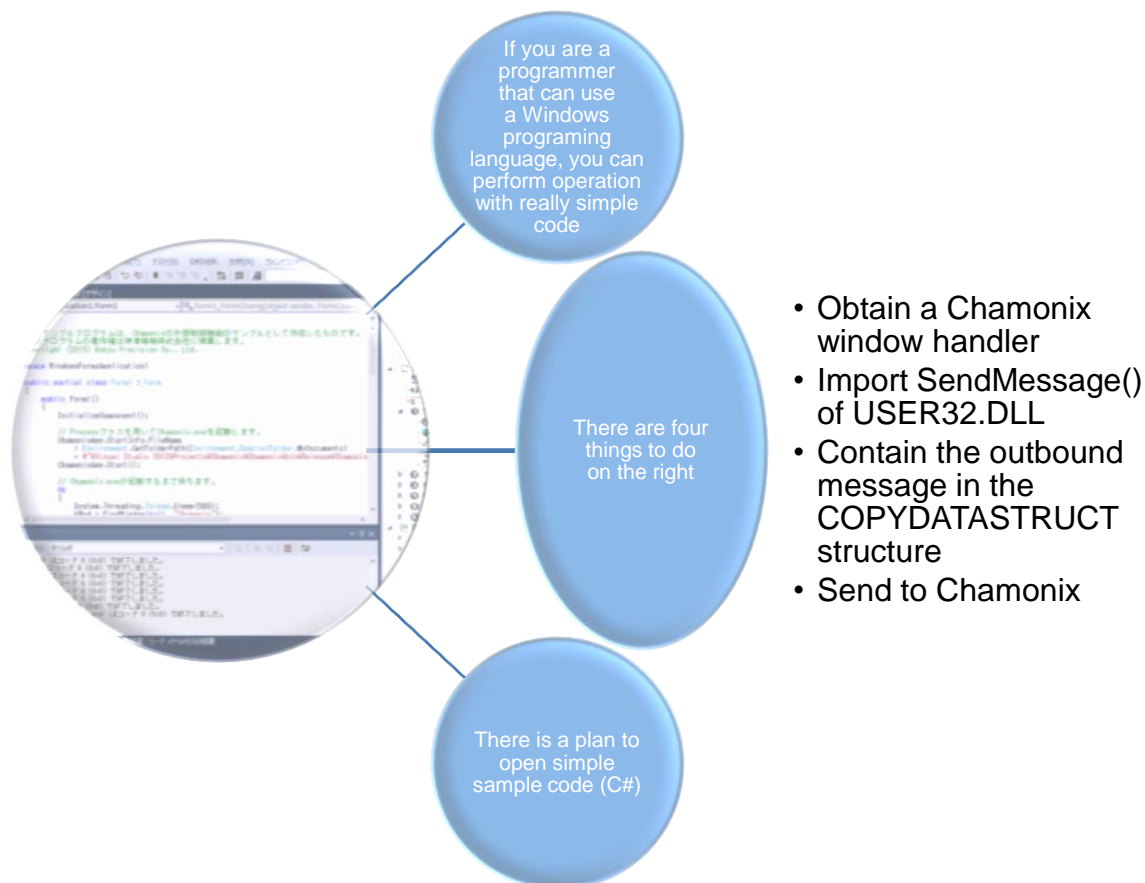
For example, refer to the Microsoft Web site for Windows API.

It supports APS, RPS, ORG, and STP remote commands.

The format is in accordance with that of ARIES remote commands.

Parameters of a response method are omitted.

15.2. Control from External Applications



Chamonix is easily controlled by adding only one code of SendMessage() in an external application.

15.3. Sending the APS Command from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

[DllImport("USER32.dll")]
private void button1_Click(object sender, EventArgs e)
{
    // Obtain a Window handler of Chamonix.
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // Set a value to the structure st.
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // Send a command to Chamonix with SendMessage.
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// Structure sent by SendMessage.
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the `SendMessage()` function²³ of Windows API from external applications.

²³ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the `SendMessage` function.

The structure of the SendMessage() function is as follows:

SendMessage (HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA (0x004A)
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT ²⁴) lParam.dwData: 0 lParam.cbData: size of sending character string (for example, 12) lParam.lpData: Sending character string (for example, APS1/9/10000)

When you want to perform deciding position of PM1 axis with the No.9 speed table to the 10000pls position using absolute position control for Chamonix, enter 12 in cbData and APS1/9/10000 in lpParam.

Parameters of a response method are omitted.

²⁴ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

15.4. Sending the RPS Command from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

[DllImport USER32.dll]

private void button1_Click(object sender, EventArgs e)
{
    // Obtain a Window handler of Chamonix.
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // Set a value to the structure st.
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // Send a command to Chamonix with SendMessage.
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// Structure sent by SendMessage.
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the `SendMessage()` function²⁵ of Windows API from external applications.

²⁵ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the `SendMessage` function.

The structure of the SendMessage() function is as follows:

SendMessage (HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA (0x004A)
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT ²⁶) lParam.dwData: 0 lParam.cbData: size of sending character string (for example, 12) lParam.lpData: Sending character string (for example, RPS1/9/100000)

When you want to perform deciding position of PM1 axis with the No.9 speed table to the 10000pls position using relative position control for Chamonix, enter 12 in cbData and RPS1/9/10000 in lpParam.

Parameters of a response method are omitted.

²⁶ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

15.5. Sending the STP Command from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

[DllImport("USER32.dll")]
private void button1_Click(object sender, EventArgs e)
{
    // Obtain a Window handler of Chamonix.
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // Set a value to the structure st.
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // Send a command to Chamonix with SendMessage.
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// Structure sent by SendMessage.
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the `SendMessage()` function²⁷ of Windows API from external applications.

²⁷ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the `SendMessage` function.

The structure of the SendMessage() function is as follows:

SendMessage (HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA (0x004A)
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT ²⁸) lParam.dwData: 0 lParam.cbData: size of sending character string (for example, 7) lParam.lpData: Sending character string (for example, STP1/1)

When you stop the PM1 axis in an emergency for Chamonix,
enter 7 in cbData
and STP1/1 in lpParam.

²⁸ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

15.6. Sending the ORG Command from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

[DllImport("USER32.dll")]
private void button1_Click(object sender, EventArgs e)
{
    // Obtain a Window handler of Chamonix.
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // Set a value to the structure st.
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // Send a command to Chamonix with SendMessage.
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}
/// <summary>
/// Structure sent by SendMessage.
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the `SendMessage()` function²⁹ of Windows API from external applications.

²⁹ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the `SendMessage` function.

The structure of the SendMessage() function is as follows:

SendMessage (HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA (0x004A)
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT ³⁰) lParam.dwData: 0 lParam.cbData: size of sending character string (for example, 7) lParam.lpData: Sending character string (for example, ORG1/9)

When you return the PM1 axis to origin with the No.9 speed table for Chamonix, enter 7 in cbData and ORG1/9 in lpParam.

³⁰ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

15.7. Canceling Standby of Programmable Grid from External Applications

```
IntPtr hWnd;

private int WM_COPYDATA = 0x004A;

[DllImport("USER32.dll")]
private void button1_Click(object sender, EventArgs e)
{
    // Obtain a Window handler of Chamonix.
    hWnd = FindWindow(null, "Chamonix");

    COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

    // Set a value to the structure st.
    copyStruct.dwData = (IntPtr)0;
    copyStruct.cbData = (uint)textBox1.Text.Length + 1;
    copyStruct.lpData = this.textBox1.Text;

    // Send a command to Chamonix with SendMessage.
    SendMessage((IntPtr)hWnd, WM_COPYDATA, this.Handle, ref copyStruct);

    //Marshal.FreeCoTaskMem(lp);
    textBox1.Text = null;
}

/// <summary>
/// Structure sent by SendMessage.
/// </summary>
private struct COPYDATASTRUCT
{
    public IntPtr dwData;
    public uint cbData;
    public string lpData;
}
```

Commands are passed to window procedures of Chamonix directly by using the `SendMessage()` function³¹ of Windows API from external applications.

³¹ See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the `SendMessage` function.

The structure of the SendMessage() function is as follows:

SendMessage (HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)	
hWnd	Enter a handle of Chamonix (main window)
Msg	WM_COPYDATA (0x004A)
wParam	Window handle to pass a message from Chamonix
lParam	Sending command (COPYDATASTRUCT ³²) lParam.dwData: 0 lParam.cbData: size of sending character string lParam.lpData: Sending character string

When a standby state is occurring with the SBY command during running of programmable grid for Chamonix, the standby state will be canceled if the string of Message= specifier agree with the string of lParam on the received command.

³² See the MSDN Web site of Microsoft Corporation for more information. Please refrain from inquiring about the COPYSTRUCT structure.

15.8. Sample Program for External Control

```

public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();

        // Start Chamonix.exe with the Process class.
        ChamonixApp.StartInfo.FileName
            = Environment.GetFolderPath(Environment.SpecialFolder.CommonProgramFilesX86)
            + @"\KOHZU\Chamonix\Chamonix.exe";
        ChamonixApp.Start();

        // Wait until Chamonix.exe starts.
        do
        {
            System.Threading.Thread.Sleep(500);
            hHnd = FindWindow(null, "Chamonix");
        }
        while (hHnd.ToInt32() == 0);
    }

    Process ChamonixApp = new Process();
    IntPtr hHnd;

    private int WM_COPYDATA = 0x004A;
    #region Import USER32.dll
    [DllImport("USER32.dll")]
    private static extern IntPtr SendMessage(IntPtr hWnd, int Msg, IntPtr wParam, ref COPYDATASTRUCT lParam);
    [DllImport("USER32.dll")]
    static extern IntPtr FindWindow(string lpClassName, string lpWindowName);
    #endregion

    private void button1_Click(object sender, EventArgs e)
    {
        // Obtain a Window handler of Chamonix.
        hHnd = FindWindow(null, "Chamonix");

        COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();

        // Set a value to the structure st.
        copyStruct.dwData = (IntPtr)0;
        copyStruct.cbData = (uint)textBox1.Text.Length + 1;
        copyStruct.lpData = this.textBox1.Text;

        // Send a command to Chamonix with SendMessage.
        SendMessage((IntPtr)hHnd, WM_COPYDATA, IntPtr.Zero, ref copyStruct);

        // Marshal.FreeCoTaskMem(lp);
        textBox1.Text = null;
    }

    /// <summary>
    /// Structure sent by SendMessage.
    /// </summary>
    private struct COPYDATASTRUCT
    {
        public IntPtr dwData;
        public uint cbData;
        public string lpData;
    }

    protected override void WndProc(ref Message m)
    {
        if (m.Msg.Equals(WM_COPYDATA))
        {
            try
            {
                COPYDATASTRUCT copyStruct = new COPYDATASTRUCT();
                copyStruct = (COPYDATASTRUCT)m.GetParam(typeof(COPYDATASTRUCT));
                string text = copyStruct.lpData;
            }
            catch
            {
                throw;
            }
            textBox1.Text = copyStruct.lpData;
        }
        base.WndProc(ref m);
    }

    private void Form1_FormClosing(object sender, FormClosingEventArgs e)
    {
        ChamonixApp.CloseMainWindow();
    }
}

```

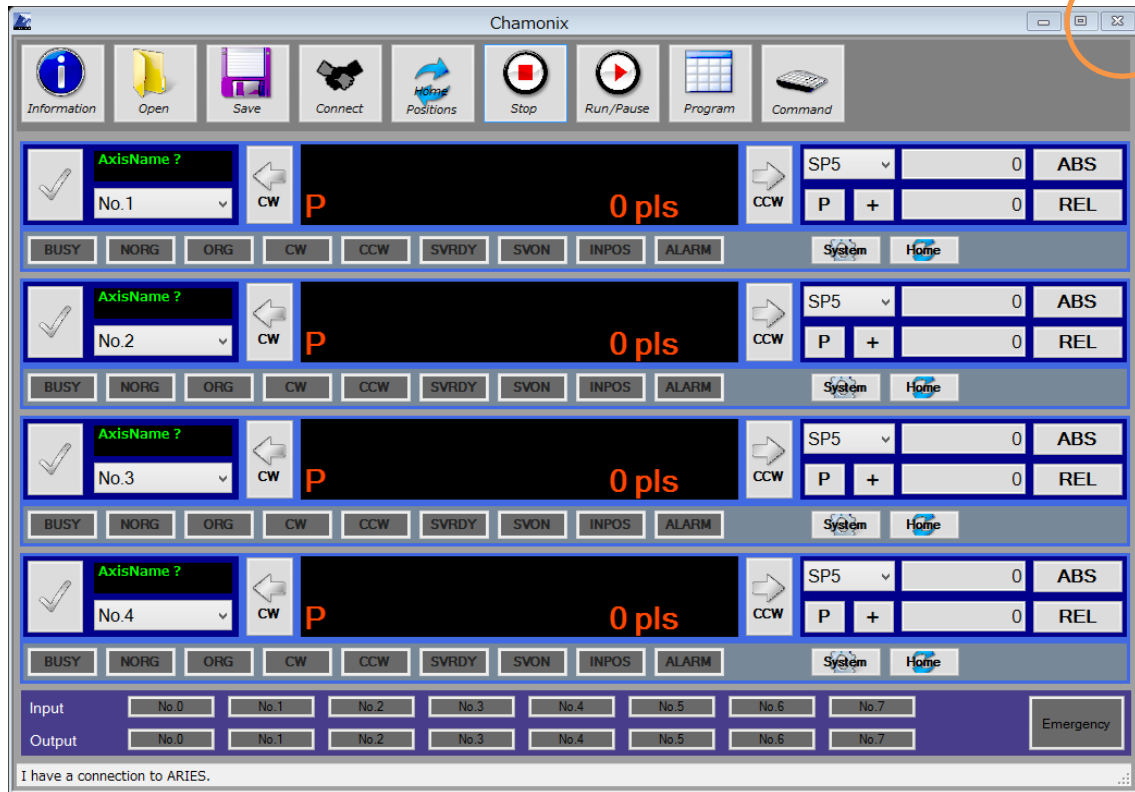
Sample code of an external application using SendMessage will be offered in C# at a Web site. The sample code is not designed to be suitable to practical use by incorporating itself. This is only an example for use of SendMessage.

Our company assumes no responsibility for customer's use of the sample code.

Copyrights in and to the sample code are owned by Kohzu Precision Co., Ltd.

16. Exiting from Chamonix

16.1. Exiting from Chamonix



Click the close box to exit from Chamonix.

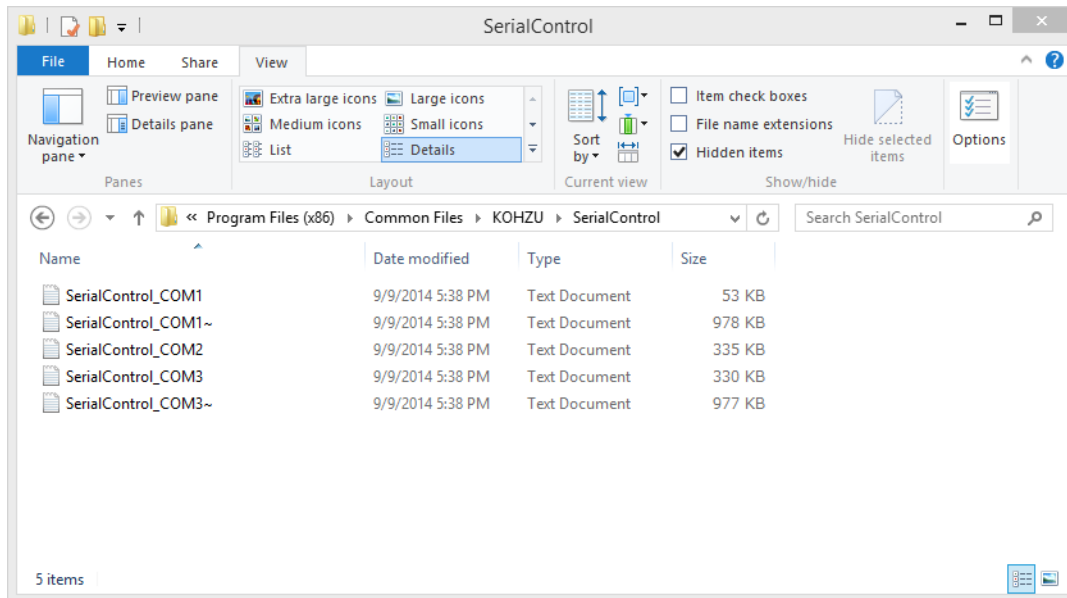
Parameters at time of the exiting are written into Config.bin of Common Files in exiting. The COM port, baud rate, conversion coefficient, name, and others are saved. System parameters of the controller give setting values of the controller priority over saved values. It starts with conditions saved in this Config.bin unless starting from a start-up file.

The STP command is sent to all the axes of the controller when Chamonix exits.

When Chamonix is shut down during control of drive with Chamonix, the drive is meant to be canceled.

17. Others

17.1. Folder to save Communication Logs



Communication logs for the controller remain in C:\Program Files (x86)\Common Files\KOHZU\SerialControl³³ folder.

Logs are backed up only one time and reset when they exceed 1 MB.

This backup is overwritten when the next backup is performed.

³³ For 32-bit OS, the logs are in C:\Program Files \Common Files\KOHZU\SerialControl.

17.2. Checking Communication Logs

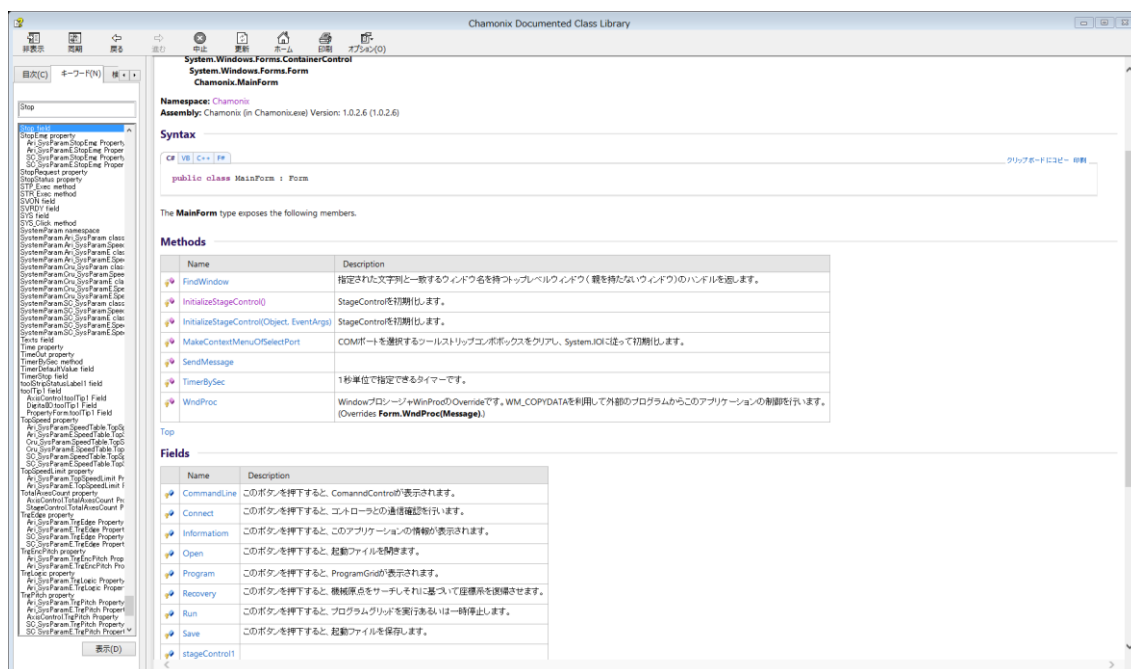
Timestamp	Direction	Data	STR1	C	RIN	RDP1	RDE1	STR2	C	RIN	RDP2	RDE2	STR3	C	RIN	RDP3	RDE3	STR4	C	RIN	RDP4	RDE4
2014/06/09 19:14:03.081	←	STR1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.089	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.109	←	RIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.109	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.145	←	RDP1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.149	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.165	←	RDE1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.169	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.241	←	STR2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.249	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.269	←	RIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.269	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.289	←	RDP2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.289	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.369	←	STR3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.369	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.389	←	RIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.389	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.409	←	RDP3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.409	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.481	←	STR4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.489	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.509	←	RIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.509	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.529	←	RDP4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.529	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.605	←	STR1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.609	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.629	←	RIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.629	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.649	←	RDP1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.649	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.665	←	RDE1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.669	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.741	←	STR2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.749	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.769	←	RIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.769	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.789	←	RDP2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.789	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.861	←	STR3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.869	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.889	←	RIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.889	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.909	←	RDP3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.909	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.981	←	STR4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:03.989	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:04.009	←	RIN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:04.009	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:04.029	←	RDP4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:04.029	→	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2014/06/09 19:14:04.101	←	STR1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

This log file contains records of transmission to and reception from a controller.

Transmission and reception time is recorded in milliseconds. The time is not accurate because of using the system clock.

It is expected that troubleshooting will become more efficient by analyzing this file when a trouble is arising.

17.3. Detailed Help Files



Detailed HTML help files are included.

Chamonix and each item of in-use libraries are written in the help files.

Press the F1 key to start the help file.

Index

- .cmxp, 32
- 1st Parameter, 65, 66, 67, 68, 69, 70, 71
- 2nd Parameters, 65, 66, 67, 70, 71
- 3rd Parameters, 65, 66, 67, 71
- ABS, 48, 51, 66
- Absolute Position, 47, 48, 50, 65, 66, 78
- Absolute Position Control, 47, 48, 50, 66, 78
- Accept, 13, 15
- Add or Remove Programs, 14
- Administrator Privileges, 12, 17, 23, 36
- API, 71, 75, 77, 79, 81, 83, 85
- Application, 8, 10, 22, 34, 56, 76, 77, 79, 81, 83, 85, 87
- ARIES, 27, 28, 29, 35, 42, 52, 53, 58, 68, 71, 75
- Axis Name, 43
- Baud Rate, 30, 31, 32, 34, 35, 88
- BUSY indicator, 48, 50
- CCW, 46
- CCW direction, 46
- Chamonix, 8, 11, 13, 14, 18, 19, 21, 22, 23, 30, 33, 54, 55, 58, 62, 71, 73, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 88, 91
- Code Signing Certificate, 17
- COM port, 30, 31, 34, 35, 36, 88
- Combo Box, 42, 45
- Command Control, 73, 74
- Communication Logs, 89, 90
- Communications Settings, 24, 25, 27, 28, 30, 31
- Compatibility, 55, 58
- Confirming and Accepting License Agreement, 15
- Confirming License Agreement, 13
- Connect button, 34, 35
- Connection, 26, 29
- Connection Check, 34, 35
- Control Display, 34, 35
- Control Panel, 14, 20, 23, 36
- Control Resolution, 47, 49
- Controller, 8, 23, 34, 35, 36, 39, 45, 52, 55, 56, 57, 58, 61, 68, 70, 73, 74, 88, 89, 90

Copyright, 87	Home Positions, 39
CRUX, 23, 24, 25, 26, 35, 42, 54, 58	icon, 30
CSV, 58, 64	Information button, 37
Current Position Display, 44	Installation, 11, 12, 13, 14, 16, 17, 18, 19, 20, 21, 22, 30, 33
Current Position of the Encoder, 44	Installation CD, 12
Current Position of the Motor Pulse, 44	KOHZU Precision, 87
CW, 46	License Agreement, 10, 13, 15, 16
CW direction, 46	LYNX, 27, 29, 35, 42
DAT, 64, 75, 78, 80, 82, 84, 86	Microsoft .NET Framework, 11
Deciding Position, 47, 48, 49, 50, 66, 67, 78, 80	Microsoft .NET Framework Full 4.0 or later, 11
Default, 30, 31, 33, 47, 49, 55, 57, 58, 64, 65, 66, 67	Microsoft Windows 8.1, 14, 19, 20
Device Manager, 23, 36	Microsoft.NET4.0, 11
Drive Automatically, 61	Modern UI, 31
Emergency, 52	Motor pulse, 44, 47, 49, 52
Emergency Stop, 52, 53, 54, 63, 82	Multi-axis drive, 51
End, 61	Open, 58
Entering, 15, 16, 17, 43, 47, 49, 56, 61, 65, 66, 67, 70, 71, 73, 74, 78, 80, 82, 84, 86	Operation Manual for Controller, 34, 45, 74
External Equipment, 26, 29	Organization, 16
Finish, 18	Parameter Setting, 55
General-Purpose Output, 68	Parity, 24, 27
Help File, 91	PAUSE, 62
Home, 39, 40	Print, 15, 59

PRINT, 59	Stop bit, 24, 27
Program Files(x86), 11	Supported OS, 11
Programmable Grid, 60, 61, 65, 72, 85, 86	System, 22, 55, 56
REL, 50, 51, 67	System Parameter, 39, 45, 47, 49, 55, 56, 58, 65, 66, 67, 88
REL button, 50	Terms of the License Agreement, 15
Relative Position, 49, 50, 67, 80	Uninstallation, 20, 21
Relative Position Control, 49, 50, 67, 80	Used Last Time, 30, 33
Repeat, 61	User Account Control, 17
Returning to Origin, 39, 40, 41, 65, 70, 84	User Information, 15, 16, 17
Returning to Origin for All Axes, 39	User's Guide, 8
Right-click, 34	Version Information, 37
Rotary Switch, 24, 25, 27, 28	Virtual COM Port, 23
RS-232C, 23, 24, 27, 34, 74	Windows7®, 11
RUN, 62	Windows8®, 11
Setup, 12, 13	Connection, 26, 29
Setup.exe, 13	Wiring, 23
Speed Table, 45, 55, 65, 78, 80, 84	Word length, 24, 27
StageControl, 22, 35	x64, 11
Start, 61	x86, 11, 19, 22, 33, 89
Start-up File, 32, 43, 88	XML, 58, 64
Status, 40, 70	Installation, 11
STOP, 63	

Revision History

Date	Version	Others
June 30, 2014	Rev. 1.0.0	
October 30, 2014	Rev.1.0.1	
December 25, 2014	Rev.1.0.2	

Section for recording

Purchased Date

Purchased from

--

Person in charge

TEL

--

Production No.

--	--

Special note

[illegible]



Headquarters

215-8521

2-6-15 Kurihira Aso-ku, Kawasaki City
Kanagawa

Tel: **+81-44-981-2130**

Fax: **044-981-2181**

E-mail: sale@kohzu.co.jp

Web Site: <http://www.kohzu.co.jp/>

Osaka branch

532-0004

2-7-38 Nishi Miyahara
Yodogawa-ku, Osaka City

Shin Osaka Nishiura Bldg. 202

Tel: **06-6398-6610**

Fax: **06-6398-6620**